THÈSE DE DOCTORAT DE

L'UNIVERSITÉ BRETAGNE SUD

COMUE UNIVERSITÉ BRETAGNE LOIRE

ÉCOLE DOCTORALE N˚601

MAThématiques et Sciences et Technologies de l'Information et de la Communication
Spécialité : Informatique

par

Nicolas AUDEBERT

intitulée

Classification de données massives de télédétection

Thèse présentée et soutenue à Palaiseau, le 17 octobre 2018,
préparée à l'Institut de recherche en informatique et systèmes aléatoires (UMR 6074),
et l'Office national d'études et de recherches aérospatiales.

Thèse n° : 502

Rapporteurs avant soutenance :

Jocelyn CHANUSSOT	Professeur, Institut polytechnique de Grenoble – GIPSA-Lab
Vincent LEPETIT	Professeur, Université de Bordeaux – LaBRI

Composition du jury :

Élisa FROMONT	Professeure, Université de Rennes 1 – IRISA	Présidente
Jocelyn CHANUSSOT	Professeur, Institut polytechnique de Grenoble – GIPSA-Lab	Rapporteur
Vincent LEPETIT	Professeur, Université de Bordeaux – LaBRI	Rapporteur
Patrick PÉREZ	Directeur scientifique, Valeo.ai – Paris	Examinateur
Yuliya TARABALKA	Chargée de recherche HDR, Inria – Sophia Antipolis	Examinatrice

Directeur de thèse

Sébastien LEFÈVRE	Professeur, Université Bretagne-Sud – IRISA

Encadrant

Bertrand LE SAUX	Chercheur, ONERA Palaiseau – DTIS

Titre : Classification de données massives de télédétection

Mots clés : apprentissage profond, télédétection, segmentation sémantique, cartographie, réseaux de neurones

Résumé : L'observation de la Terre permet de modéliser et de comprendre son évolution. L'abondance d'images de télédétection aériennes et satellitaires nécessite la mise en œuvre de moyens d'analyse automatiques, capables d'interpréter ces données et de cartographier la surface du globe. Cette thèse traite de la conception, du déploiement et de la validation de stratégies d'apprentissage automatique, en particulier de réseaux de neurones convolutifs profonds, pour la compréhension d'images et la cartographie automatisée. Nous proposons des modèles pour l'interprétation d'images couleur, multispectrales et hyperspectrales, capables de prendre en compte les interactions spatiales entre entités géométriques et produisant des cartes d'une précision permettant la détection d'objets. Nous introduisons des architectures de fusion de données par apprentissage multi-modal et correction résiduelle afin de tirer parti des données ancillaires, comme les modèles numériques de terrain et les connaissances géographiques disponibles a priori. Enfin, nous étudions les capacités de généralisation de ces modèles dans des cas extrêmes de jeux de données limités ou massifs. Nous validons tout au long de cette thèse nos contributions sur de multiples jeux de données aériens et satellitaires pour la classification des sols et de leurs usages, l'extraction de bâtiments et la détection de véhicules.

Titre : Classification of big remote sensing data

Mots clés : deep learning, remote sensing, semantic segmentation, neural networks, mapping

Résumé : Earth Observation allows us to modelize and understand the evolution of our planet. The profusion of aerial and satellite remote sensing images induces the need for automated tools able to semantize such raw data in order to map the Earth. This thesis studies the design, implementation and validation of machine learning strategies, specifically deep convolutional neural networks, for image understanding and automatic mapping. We introduce models for automated interpretation of color, multispectral and hyperspectral images, that are able to exploit spatial relationships between geometrical entities and to produce high precision maps relevant for object detection. We design data fusion architectures using multimodal learning and residual correction that can leverage ancillary data, such as digital surface models and prior geographical knowledge. Finally, we study the generalization abilities of those networks for extreme cases of both limited and very large datasets. All along this work, we thoroughly validate our contributions on various aerial and satellite datasets for land cover and land use classification, building footprints extraction and vehicle detection.

Remerciements

Préparer une thèse est une aventure personnelle, mais pas le moins du monde solitaire. [1]

En premier lieu, je tiens à remercier vivement Prof. Jocelyn Chanussot et Prof. Vincent Lepetit pour leurs retours inestimables comme rapporteurs de ce manuscrit. À eux, ainsi qu'au reste du jury constitué de Prof. Élisa Fromont, Dr. Patrick Pérez et Dr. Yuliya Tarabalka, j'adresses mes plus sincères remerciements pour le temps qu'ils m'ont consacré. C'est une grande fierté pour moi que mes travaux aient trouvé grâce à leurs yeux.

Je souhaite ensuite exprimer toute ma reconnaissance envers mes encadrants. Sébastien, ta rigueur et ton attention au détail sont une source perpétuelle d'admiration, la cohérence scientifique de ces travaux te doit beaucoup. Bertrand, j'espère qu'un peu de ton optimisme infatigable et ton éthique remarquable auront déteint sur moi ! Je ne saurais vous remercier assez tous les deux pour les encouragements constants que vous m'avez prodigué, votre patience lors des relectures de ce manuscrit et votre présence attentive qui a rendu cette thèse particulièrement agréable. [2]

Je me dois également de saluer la disponibilité et la sympathie d'Alexandre Boulch et d'Adrien Chan-Hon-Tong, qui m'ont accueilli plus que de raison dans leur bureau et ont toujours pris le temps de dispenser conseils avisés et réponses à mes questions diverses et variées. Certains des résultats présentés dans les chapitres qui suivent leur doivent beaucoup ! [3]

J'ai eu en outre la chance d'être suivi durant cette thèse par non pas un, mais bien deux (!) comités scientifiques d'une grande qualité. Je remercie notamment pour leurs précieux conseils Ronan Fablet, Nicole Vincent, Xavier Ceamanos, Véronique Achard, Peppino Terpollili, Nooman Keskes, Georges Oppenheim et Dominique Dubucq.

À l'ONERA, je remercie chaleureusement mes co-bureaux successifs, Élyse, Emannuelle et Rodolphe, et la cohorte de doctorants passés et présents [4] – Calum, David C., Flora, Guillaume, Hélène, Isabelle, Joris, Maxime D., mais aussi David S., Florent, Marcela, Maxime B., Maxime F., Pierre, Rodrigo, Soufiane – qui ont rendu cette expérience conviviale et festive, sans oublier nos formidables (ex-)stagiaires et (ex)-apprentis – Anthelme, Benjamin, Guillaume, Juliette, Martin, Oriane, Robin, Sémy, Simon, Thibault, Xavier – dont certains qui continuent en thèse et à qui je souhaite bien du courage ! Et comme ils méritent bien une phrase entière à eux seuls, je salue avec amitié Javiera, qui a rendu la section sur MiniFrance possible, et Hicham, dont la longévité légendaire m'a permis de le côtoyer bien plus que je n'aurais su l'espérer. Je remercie en outre l'ensemble de l'équipe IVA, dont la culture générale et scientifique enrichit les discussions de salle de pause et les transforme en petits moments d'émerveillements entrecoupant les laborieuses journées de travail. Enfin, je suis redevable à Claire, Fabrice, Florence et Françoise pour leur inestimable aide administrative.

À Vannes, je tiens à remercier l'ensemble de l'équipe OBELIX [5] dont l'accueil toujours chaleureux a rendu mes visites on ne peut plus plaisantes, et m'a fait regretter de ne pas y passer plus souvent !

Hors des murs des labos, je me remercie du fond du cœur la ribambelle d'ami(e)s dont la simple présence est une source de joie inépuisable et tout particulièrement Francis, Michel, Rémi, Sélim, Timothée [6], Vérène et l'ensemble des tetracontakaidiens : Adèle, Aurore,

1. Une fois n'est pas coutume, cette page contiendra bien plus de notes de bas de page que nécessaire.
2. Non pas que ces trois ans fûrent une promenade de santé, mais les heurts et les accrocs ne sont jamais demeurés bien longtemps, signe d'un encadrement d'une grande qualité.
3. Et, je le pense sincèrement, vous avez tous les deux été des encadrants non-officiels dont je n'aurais pas pu me passer !
4. À ceux qui restent, je peux en témoigner : on finit vraiment par en voir le bout !
5. Je n'en fais pas la liste car j'ai trop peur d'un oubli !
6. Et aux jeudi sushi, moment phare de la semaine s'il en est.

Clément, Étienne, Gabriel, Guillaume, Johan, Jolan, Léni, Loïc, Paul, Pierre et Stéphane. [7]

Enfin, je ne remercierai jamais assez mes parents et ma sœur [8] pour leur soutien inconditionnel toutes ces longues années. Cette thèse est un petit peu la vôtre. À l'ensemble de ma famille, qui, sans avoir toujours compris de quoi il en retournait [9], n'a jamais cessé de s'intéresser à mes activités, merci ! Et enfin, à Camille, tu me fais chaque jour le plus grand des cadeaux en étant dans ma vie.

Pour terminer, quelques remerciements dans le désordre : à l'adorable Nobu pour sa photogénie [10], à Alexandra Elbakyan pour sci-hub qui m'a dépanné bien trop de fois, à Pascal Monasse pour m'avoir laissé l'opportunité d'enseigner, aux collègues croisé(e)s en conférence qui rendent les voyages plus agréables et aux contributeurs et contributrices de Wikipédia, Zotero, LaTeX et tous les développeurs et développeuses [11] qui m'ont bien facilité la vie ces dernières années.

Aux inévitables oublis, je tiens à vous assurer toute ma gratitude.

Lecteur, lectrice, bonne lecture.

7. Quel enfer pour retrouver vos prénoms !
8. Et ma future nièce, qui sait, elle lira peut-être ces lignes dans 20 ans.
9. Notez que c'est moi qui explique mal.
10. Cf. figure 2.5.
11. PyTorch, Caffe, Python, matplotlib, scikit-learn, scikit-image, numpy, scipy, tqdm, Jupyter, IPython, j'en passe et des meilleurs. Pour reprendre la formule de Bernard de Chartres, si je suis un nain juché sur des épaules de géants, ces outils ont constitué l'essentiel de mon matériel d'escalade.

Table des matières

1 Introduction **1**
- 1.1 Contexte . . . 2
- 1.2 Domaine . . . 3
- 1.3 Problématique . . . 5
- 1.4 Contributions . . . 6

2 État de l'art **9**
- 2.1 Apprentissage profond pour la vision artificielle . . . 10
- 2.2 Apprentissage profond pour la segmentation sémantique . . . 28
- 2.3 Apprentissage pour le traitement d'images de télédétection . . . 37

3 Cartographie automatisée d'images aériennes **59**
- 3.1 Classification par région d'images aériennes . . . 60
- 3.2 Réseaux de neurones profonds . . . 66
- 3.3 Évaluation des modèles . . . 72

4 Extension aux capteurs non-conventionnels **91**
- 4.1 Images multispectrales . . . 92
- 4.2 Imagerie hyperspectrale . . . 98
- 4.3 Imagerie laser et modèles de terrain . . . 109

5 Segmentation sémantique multimodale **119**
- 5.1 Apprentissage multimodal . . . 120
- 5.2 Fusion de modèles . . . 123
- 5.3 Connaissances *a priori* . . . 131

6 Généralisabilité des modèles **141**
- 6.1 Génération de données synthétiques . . . 142
- 6.2 Cas des données massives . . . 149

7 Spatialisation des prédictions pixelliques **159**
- 7.1 *Segment-before-detect* . . . 160
- 7.2 Segmentation sémantique par régression des cartes de distances . . . 171

8 Conclusion et perspectives **187**

A Jeux de données **I**
- A.1 Jeux de données en télédétection . . . I
- A.2 Jeux de données en interprétation de scènes . . . VII

B Code **XIII**
- B.1 FCN pour la cartographie sémantique . . . XIII
- B.2 *DeepHyperX* . . . XIII
- B.3 *MiniFrance* . . . XIII
- B.4 HyperGANs . . . XIII

C Liste des acronymes **XV**

D Glossaire **XIX**

Liste des figures

1.1 Satellites de la constellation A-Train pour l'observation de la Terre en 2018. . 2
1.2 Processus d'interprétation automatique des images d'observation de la Terre pour la cartographie. 3

2.1 Introductions de *Computing Machinery and Intelligence* (Turing, 1950) et *Summer Vision Project* (Papert, 1966). 10
2.2 Modélisation d'un neurone artificiel. 13
2.3 Perceptron à une et plusieurs couches. 14
2.4 Exemples de fonctions d'activation. 14
2.5 Exemples de filtrages par différents noyaux de convolution. 22
2.6 Opérateur de convolution et variantes sur une image 23
2.7 Sous-échantillonnage et sur-échantillonnage par valeurs maximales en deux dimensions. 25
2.8 Exemple de classification et de segmentation sur une même image. 29
2.9 Architecture LeNet-5. 29
2.10 Architecture AlexNet 30
2.11 Architecture VGG-16 31
2.12 Architecture GoogLeNet. 31
2.13 Module *Inception*. 32
2.14 Convolutions séparables en profondeur. 32
2.15 Bloc convolutif résiduel 32
2.16 Bloc convolutif dense 32
2.17 Architecture ResNet-34 33
2.18 Architecture DenseNet-121 33
2.19 AlexNet entièrement convolutif 34
2.20 L'observation de la Terre implique une grande variété de capteurs dotés de spécificités qui leur sont propres. 37
2.21 Un capteur multispectral acquiert plusieurs bandes spectrales larges réparties sur le spectre lumineux infrarouge, visible et parfois ultraviolet. 38
2.22 Un capteur hyperspectral acquiert de nombreuses bandes spectrales étroites régulièrement réparties sur sa plage d'acquisition. 38
2.23 Schéma représentatif de la différence entre MNT et MNE. 39

3.1 Cartographie automatisée d'images aériennes. 60
3.2 Segmentations d'une image naturelle. 62
3.3 Segmentations d'une image aérienne du jeu de données ISPRS Potsdam. . . . 65
3.4 Segmentation sémantique par régions d'une image aérienne. 67
3.5 Réseau de neurones entièrement convolutif – architecture SegNet. 68
3.6 Couche convolutive multinoyau. 69
3.7 Supervision profonde d'un SegNet à trois échelles. 71
3.8 Métriques de classification binaire 72
3.9 Comparaison des cartes prédites en classification par régions et classification par FCN. 74
3.10 Effets de la couche convolutive multinoyau sur des extraits du jeu de données ISPRS Vaihingen. 78
3.11 Effet de la supervision multiéchelle sur un extrait du jeu de données ISPRS Vaihingen. 79

3.12 Comparaison des segmentations obtenues sur un extrait du jeu de test ISPRS Vaihingen. 80
3.13 Cas limites de désaccord entre les prédictions faites par SegNet et la vérité terrain. 80
3.14 Carte sémantique obtenue par SegNet sur la tuile 3_11 du jeu de données ISPRS Potsdam 83

4.1 Distributions des intensités pour les canaux rouge, vert, bleu et infrarouge du jeu de données ISPRS Potsdam. 92
4.2 Cartes de corrélation entre canaux du jeu de données ISPRS Potsdam. 93
4.3 Exemples de prédictions du modèle SegNet MSI entraîné sur D2 (avec nuages). 97
4.4 Exemples de prédictions du modèle SegNet MSI entraîné sur D1 (sans nuage). 97
4.5 Exemple de cube hyperspectral sur le jeu de données *Pavia University*. 99
4.6 Exemple de réflectances caractéristiques de diverses surfaces terrestres. . . . 99
4.7 CNN unidimensionnel pour la classification de spectres. 104
4.8 Architecture hybride ACP+CNN pour la classification d'hypercubes. 105
4.9 CNN 3D pour la classification d'hypercubes. 106
4.10 Tuile 30 du jeu de données ISPRS Vaihingen selon plusieurs modalités. . . . 109
4.11 Différences entre les prédictions des modèles IRRV et composite. 112

5.1 Exemples d'architectures multimodales de réseaux profonds. 120
5.2 Architecture FuseNet. 123
5.3 Stratégies de fusion pour l'architecture FuseNet. 124
5.4 Correction résiduelle appliquée à SegNet. 125
5.5 Module de correction résiduelle. 126
5.6 Exemples de prédictions multimodales réussies sur Vaihingen. 129
5.7 Effet des stratégies de fusion sur un extrait du jeu de données ISPRS Potsdam. 130
5.8 Les erreurs dans le MNH du jeu de données ISPRS Vaihingen sont mal gérées par les deux méthodes de fusion. 130
5.9 Tuile 4_12 du jeu de données ISPRS Potsdam et données OSM correspondantes. 132
5.10 Correction résiduelle appliquée à un OSMNet et un SegNet. 133
5.11 Exemple de segmentation obtenue sur le jeu de données ISPRS Potsdam en incluant la donnée OSM. 135
5.12 Évolution des prédictions de SegNet RVB et RVB+OSM. 135

6.1 La structure de GAN utilisée pour la synthèse de spectres artificiels. 143
6.2 Spectre moyen et écart-type pour deux classes de matériaux du jeu de données Pavia Center. 145
6.3 ACP sur les spectres réels et synthétiques. 146
6.4 Interpolations dans l'espace latent des spectres 148
6.5 Cartes sémantiques prédites sur la tuile 21 de Vaihingen par SegNet entraîné sur Vaihingen et sur Potsdam. 150
6.6 Présentation du jeu de données MiniFrance. 151
6.7 Exemple de carte sémantique obtenue sur MiniFrance. 153

7.1 Illustration de la méthode *segment-before-detect* pour la segmentation, détection et classification de véhicules. 161
7.2 Processus de localisation des instances de véhicules par ouverture morphologique et extraction des composantes connexes. 161
7.3 Augmentation de données sur un véhicule de la base VEDAI. 162
7.4 Exemples d'annotations dans les jeux de données considérés. 163
7.5 Exemples de segmentations obtenues sur Potsdam et Christchurch. 164
7.6 Exemples de détections sur Potsdam et Christchurch 165

7.7 Visualisation des véhicules présents sur une des tuiles du jeu de données ISPRS Potsdam. 167
7.8 Visualisation des véhicules présents sur une des tuiles du jeu de données NZAM/ONERA Christchurch. 168
7.9 Segmentations réussies mais mauvaises classifications sur Potsdam. 170
7.10 Segmentations et classifications réussies sur Potsdam. 170
7.11 Différentes représentations de segments annotés. 172
7.12 Apprentissage multitâche (classification pixel à pixel et régression des cartes de distances). 173
7.13 Extrait des résultats de segmentation sur le jeu de données ISPRS Vaihingen. 176
7.14 Extrait des résultats de segmentation sur le jeu de données ISPRS Potsdam. . 176
7.15 Extrait des résultats de segmentation sur le jeu de données INRIA *Aerial Image Labeling*. 177
7.16 Exemple de résultats de segmentation sémantique sur le jeu de données CamVid.179
7.17 Exploration de plusieurs valeurs de λ sur le jeu de données ISPRS Vaihingen. 179

A.1 Images ortho-rectifiées et MNH pour le jeu de données ISPRS Vaihingen. . . I
A.2 Images ortho-rectifiées et MNH pour le jeu de données ISPRS Potsdam. . . . II
A.3 Images ortho-rectifiées et MNH pour le jeu de données DFC 2015. III
A.4 Données d'entraînement du concours DFC 2018. IV
A.5 Exemples d'images extraites de la base de données *Inria Aerial Image Labeling*. V
A.6 Extraits d'images annotées du jeu de données VEDAI. VI
A.7 Images et annotations extraites du jeu de données NZAM/ONERA Christchurch. VII
A.8 Images RVB (première ligne) et annotations pixelliques (deuxième ligne) extraites du jeu de données CamVid. VIII
A.9 Images RVB, cartes de profondeur et annotations extraites du jeu de données SUN RGB-D. VIII
A.10 Répartition des pixels des jeux de données ISPRS dans différentes classes d'intérêt. IX
A.11 Répartition des pixels dans les jeux de données du *Data Fusion Contest*. X

Liste des tableaux

3.1 Comparaison des algorithmes de segmentation sur le jeu de données ISPRS Vaihingen. . . . 74
3.2 Résultats de segmentation sémantique sur le jeu de validation ISPRS Vaihingen. 75
3.3 Résultats de segmentation sémantique sur le jeu de validation *International Society for Photogrammetry and Remote Sensing* (ISPRS) Vaihingen en fonction du recouvrement de la fenêtre glissante. . . . 76
3.4 Comparaison de différentes initialisations sur le jeu de validation ISPRS Vaihingen. . . . 77
3.5 Résultats de segmentation sémantique en validation sur le jeu de données ISPRS Vaihingen. . . . 78
3.6 Résultats de validation multiéchelle sur le jeu de données ISPRS Vaihingen. . 78
3.7 Résultats du ISPRS 2D *Semantic Labeling Challenge* Vaihingen (ordre chronologique). . . . 81
3.8 Résultats du ISPRS 2D *Semantic Labeling Challenge* Potsdam (ordre chronologique). . . . 81

4.1 Comparaison des performances de segmentation sémantique de SegNet sur le jeu de données de validation ISPRS Potsdam pour différentes combinaisons de canaux. . . . 93
4.2 Descriptifs des deux jeux de données d'images Sentinel-2 considérés. . . . 94
4.3 Liste des classes des jeux de données D1 et D2 dérivées de *GlobeCover* 2009. 95
4.4 Performances de SegNet sur les jeux de données D1 et D2 Sentinel-2 96
4.5 Récapitulatif des principaux jeux de données publics annotés en imagerie hyperspectrale. . . . 102
4.6 Résultats de classification de différents modèles de notre boîte à outils *DeepHyperX* sur les jeux de données Indian Pines, Pavia University et DFC 2018. 108
4.7 Résultats de validation sur le jeu de données ISPRS Vaihingen pour un modèle SegNet entraîné sur les Modèle Numérique d'Élévation (MNE) et Modèle Numérique de Hauteur (MNH) . . . 110
4.8 Résultats de validation sur le jeu de données ISPRS Vaihingen pour un modèle SegNet entraîné sur les images composites . . . 111
4.9 Résultats de validation sur le jeu de données ISPRS Potsdam pour un modèle SegNet entraîné sur les images composites . . . 111

5.1 Résultats de segmentation sémantique multimodale sur le jeu de validation ISPRS Vaihingen. . . . 126
5.2 Résultats de segmentation sémantique multimodale sur le jeu de test ISPRS Vaihingen (approches multimodales). . . . 127
5.3 Résultats de segmentation sémantique multimodale sur le jeu de test ISPRS Potsdam (approches multimodales). . . . 127
5.4 Résultats de segmentation sémantique multimodale avec OSM sur le jeu de données ISPRS Potsdam . . . 134

6.1 Exactitudes d'une SVM linéaire appliquée sur les spectres réels et synthétiques (Pavia University). . . . 147
6.2 Scores d'exactitudes obtenus par un classifieur entièrement connecté à 4 couches sur plusieurs jeux de données en utilisant différentes augmentations de données. . . . 149

6.3 Résultats de segmentation sémantique et d'apprentissage par transfert pour le jeu de données ISPRS Vaihingen. 150
6.4 Liste des villes présentes dans MiniFrance. 152
6.5 Taxonomie des types d'occupation des sols de UrbanAtlas 2012. 154
6.6 Performances de segmentation sémantique d'un modèle SegNet sur MiniFrance 154
6.7 Comparaison des statistiques au niveau pixel entre Vaihingen et MiniFrance. 155

7.1 Nombre de véhicules par classe dans les différents jeux de données. 163
7.2 Résultats de segmentation sémantique sur le jeu de données ISPRS Potsdam à 12,5 cm/px 164
7.3 Résultats de segmentation sémantique sur NZAM/ONERA Christchurch . . 165
7.4 Segmentation d'instance et détection de véhicules pour différents prétraitements morphologiques 166
7.5 Résultats de détection de véhicules sur Potsdam et Christchurch. 166
7.6 Erreur moyenne d'estimation du nombre de véhicules par cellule de $125\,\mathrm{m}^2 \times 125\,\mathrm{m}^2$. 167
7.7 Résultats de classification de plusieurs CNN sur VEDAI (en %). 169
7.8 Résultats de classification d'AlexNet sur VEDAI avec plusieurs prétraitements (en %). 169
7.9 Résultats de classification de véhicules sur les vérités terrain augmentées de Potsdam et Christchurch. 171
7.10 Résultats de validation croisée sur les jeux de données ISPRS (multitâche). . 174
7.11 Performances en extraction de bâtiments sur le jeu de données INRIA *Aerial Image Labeling*. 177
7.12 Résultats sur le jeu de données SUN RGB-D (images de 224×224px). 177
7.13 Résultats de segmentation sémantique sur le jeu de données DFC 2015 . . . 178
7.14 Performances de segmentation sémantique sur le jeu de données CamVid. . . 178



Chapitre

1 Introduction

[The Computer] was the first machine man built that assisted the power of his brain instead of the strength of his arm.

— Grace Hopper

Sommaire

1.1 Contexte 2
1.2 Domaine 3
1.2.1 Images de télédétection 4
1.2.2 Apprentissage statistique 4
1.2.3 Vision par ordinateur 5
1.3 Problématique 5
1.4 Contributions 6

1.1 Contexte

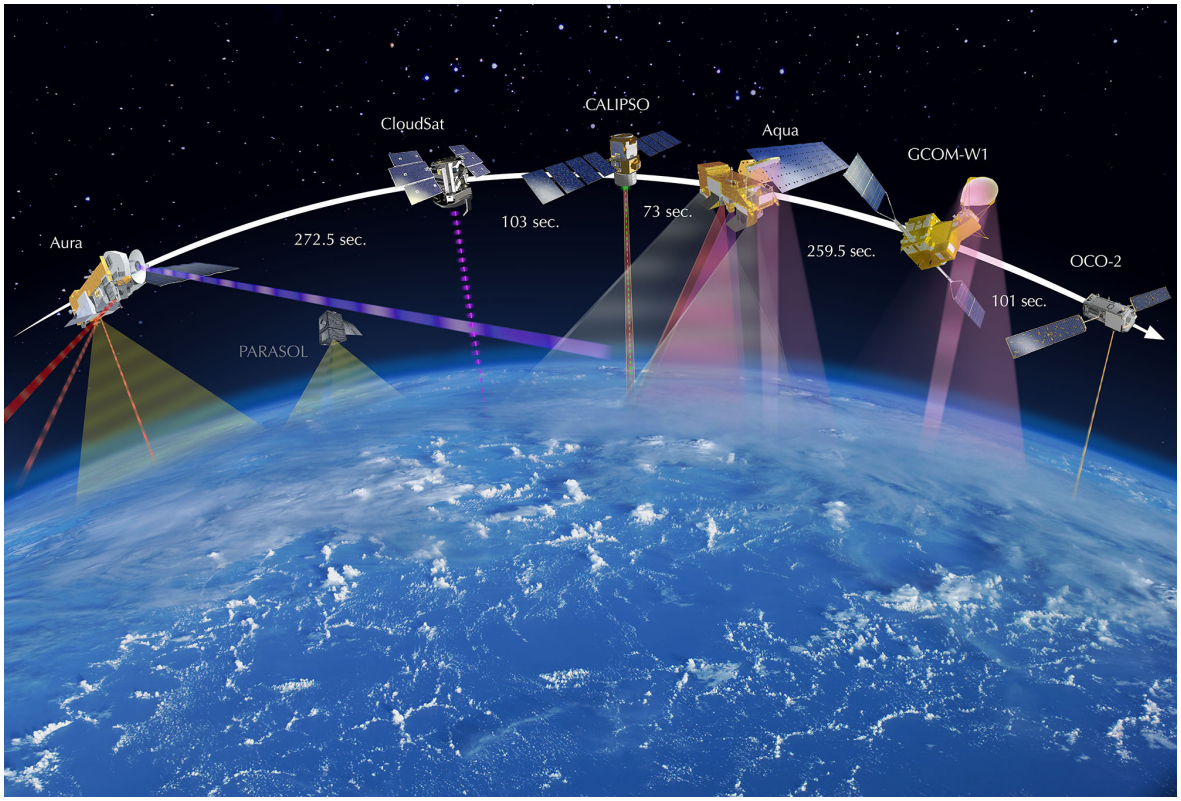


Figure 1.1 – Satellites de la constellation A-Train pour l'observation de la Terre en 2018.
Crédits image : NASA JPL (domaine public)

Toute démarche scientifique débute par une observation. Comprendre un objet passe par un examen attentif des phénomènes qu'il engendre et notre planète ne fait pas exception à cette logique. Ce n'est donc pas une surprise si, lors de l'avènement des premiers programmes spatiaux, les premiers satellites mis en orbite étaient résolument tournés vers la Terre. En effet, l'altitude a permis à la communauté scientifique d'acquérir une toute nouvelle perspective.

L'imagerie aérienne et satellitaire est un outil désormais omniprésent dans les sciences modernes. Comprendre la Terre est un enjeu scientifique majeur, et l'observation est le premier pas nécessaire à toute tentative de modélisation. Qu'il s'agisse de météorologie, d'océanographie, d'écologie ou de géographie, les images de télédétection fournissent une information formidablement riche.

L'intensification des efforts pour imager la Terre dans son entièreté le plus souvent possible n'a donc rien d'étonnant. Des constellations de satellites telles que Landsat, SPOT (Satellites Pour l'Observation de la Terre), Sentinel ou le A-Train (cf. Figure 1.1) survolent le globe en continu. Les satellites Sentinel-2A et 2B acquièrent à eux seuls plus de 6 To de données chaque jour et imagent l'intégralité de la planète chaque semaine. Pour autant, exploiter cette masse de données n'est pas chose aisée. Interpréter et comprendre une image satellite nécessite une expertise spécifique, mêlant connaissance de la physique du capteur et maîtrise du champ applicatif considéré.

De nombreux domaines peuvent cependant bénéficier de la fouille systématique des données d'observation de la Terre :

— **Écologie** : études de santé des espaces forestiers (déforestation, pollution), suivi des icebergs et évaluation de la fonte des glaces, détection précoce des dégazages pétroliers et phénomènes de marée noire…
— **Météorologie** : anticipation et suivi des phénomènes météorologiques intenses (tempêtes, cyclones), évaluation des effets liés au réchauffement climatique…

— **Urbanisme** : évaluation de l'expansion urbaine et du maillage routier, planification de l'intervention des secours après un séisme, identification des îlots de chaleur…
— **Législation** : contrôle des lois concernant les cycles agricoles, surveillance de l'apparition de bâti non autorisé, détection de navires de contrebande ou en situation de pêche illégale…

Malgré les efforts des acteurs institutionnels comme l'Institut national de l'information géographique et forestière (IGN) ou le Centre national d'études spatiales (CNES), les photo-interprètes ne peuvent assumer seuls cette responsabilité. L'automatisation présente alors une alternative intéressante. Conférer aux machines la capacité d'interpréter les images de la Terre permettrait de multiplier les observations, pour en tirer à la fois informations et modèles. Pour ce faire, il convient de s'appuyer sur des outils de perception artificielle adaptés à la compréhension d'images. À l'heure actuelle, l'état de l'art en vision par ordinateur repose majoritairement sur les réseaux de neurones profonds, dont les performances en classification d'images, détection d'objets et reconnaissance de formes ont permis des avancées significatives en intelligence artificielle. Un processus idéal de cartographie itératif pour l'observation de la Terre est détaillé dans la Figure 1.2.

Cette thèse s'articule ainsi de la façon suivante. Nous cherchons à concevoir, implémenter et valider des modèles de réseaux de neurones artificiels profonds pour l'interprétation automatisée d'images aériennes et satellites. Ces données peuvent être issues de capteurs multiples, sur une large variété de scènes et pour différents champs d'application.

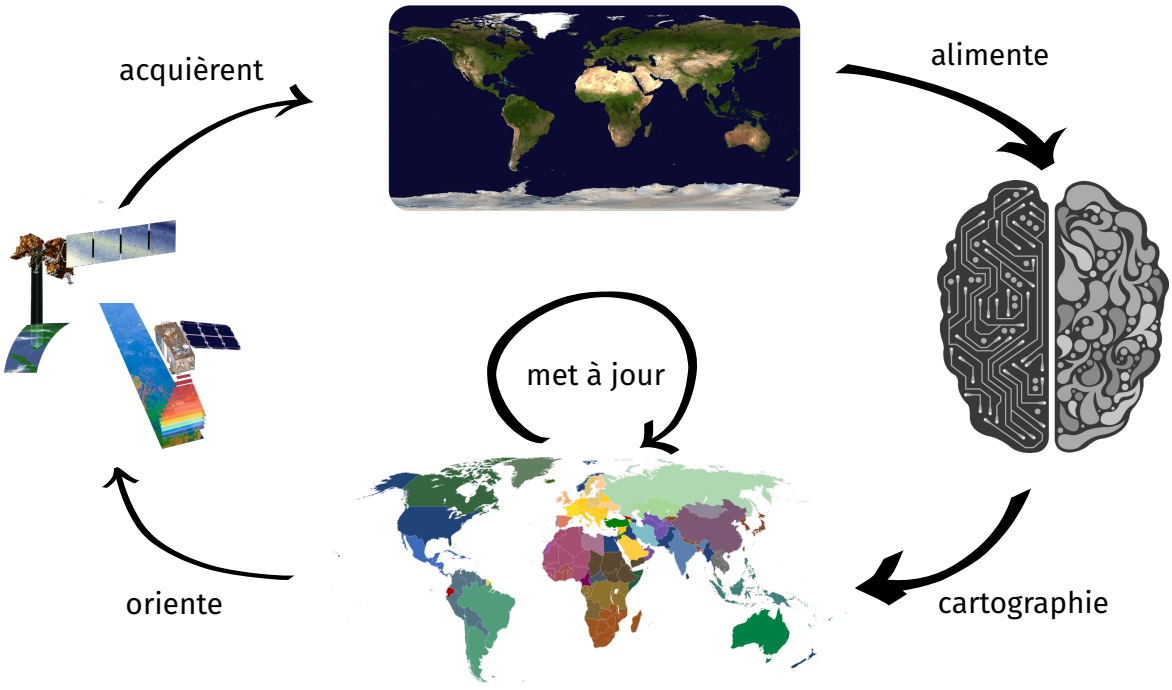


Figure 1.2 – Processus d'interprétation automatique des images d'observation de la Terre pour la cartographie.

1.2 Domaine

Cette thèse se place à l'intersection de trois domaines scientifiques : la télédétection, la vision artificielle et l'apprentissage statistique. La littérature en vision par ordinateur est abondante concernant l'interprétation de données visuelles, y compris pour la télédétection. Récemment, les méthodes dites d'apprentissage profond ont permis de réaliser des avancées considérables en interprétation automatique d'images. Toutefois, l'essentiel de la communauté s'intéresse aux tâches perceptuelles liées à des scènes de la vie quotidienne.

En particulier, les applications s'inscrivent souvent dans le cadre d'extraction d'information dans des images ou vidéos en intérieur ou en extérieur, notamment pour la domotique, la navigation autonome et le multimédia. L'interprétation d'images de télédétection bénéficie de ces recherches, mais ses spécificités en termes de point de vue et de capteurs posent par ailleurs des problèmes nouveaux.

1.2.1 Images de télédétection

Les images de télédétection regroupent une grande variété de données acquises par des moyens spatiaux ou aéroportés. Les observations idéales s'effectuent au nadir, c'est-à-dire à la verticale du sol. En pratique, il est rare que l'instrument de bord soit parfaitement orienté, notamment pour les acquisitions satellitaires. Une étape d'orthorectification est généralement introduite pour corriger les erreurs dues à l'inclinaison de la prise de vue, mais aussi au relief du terrain et aux effets de parallaxe.

Dans tous les cas, les capteurs vont mesurer l'énergie radiative émise par la scène observée. Avec les capteurs actifs, comme le radar ou le lidar, une onde électromagnétique est envoyée et la mesure porte sur celle renvoyée en retour par la scène. Ils sont ainsi leur propre source de signal. À l'inverse, les capteurs passifs mesurent soit l'énergie radiative émise par la scène (capteurs thermiques dans l'infrarouge), soit l'énergie solaire réfléchie (capteurs multispectraux). Ils nécessitent ainsi une source de lumière externe.

En dépit des spécificités des capteurs, plus nombreux et plus variés que les appareils photos et caméras grand public, le traitement des images de télédétection est proche de celui des images de la vie quotidienne. En effet, dans les deux cas, il s'agit d'extraire de l'information d'images, une tâche de perception artificielle communément appelée vision par ordinateur. Une communauté existe ainsi à la lisière entre télédétection et vision artificielle, concevant ou adaptant des algorithmes de traitement d'images pour l'observation de la Terre.

1.2.2 Apprentissage statistique

La sémantisation des images d'observation de la Terre passe nécessairement par une étape automatisée. En effet, le volume de données acquises chaque année par les capteurs aéroportés et spatiaux est simplement trop conséquent pour que des photo-interprètes humains soient en mesure de les traiter en temps réel.

L'apprentissage statistique permet de déléguer l'extraction de connaissances à une machine afin de l'automatiser. Dans la majorité des cas, il s'agit d'estimer la valeur d'un paramètre ou de prendre une décision parmi un éventail de choix. On parle dans le premier cas de régression et dans le second de classification.

Dans le cas de la photo-interprétation, l'expert humain réalise une étude des images de télédétection à sa disposition pour en réaliser une cartographie. Il s'agit d'un processus de décision, durant lequel le photo-interprète cherche des indices permettant de déclarer qu'un objet ou une région observée appartient à une catégorie précise. L'apprentissage statistique consiste donc à modéliser numériquement ce processus de classification afin de l'automatiser.

Cette modélisation passe par une phase d'apprentissage ou d'entraînement, durant laquelle le modèle consulte des exemples pour alimenter sa base de connaissances. Une fois l'apprentissage terminé, le modèle statistique est alors appliqué sur des données inédites afin de tenter de généraliser les décisions prises sur les exemples. La qualité de cette généralisation est le point critique de cette démarche et peut rencontrer deux types d'obstacle. Si le modèle contient de nombreux paramètres et est exposé à peu d'exemples, alors la base de connaissances risque d'être simplement mémorisée : on parle alors de surapprentissage. Inversement, si le modèle apprend sur de nombreux exemples mais avec trop peu de paramètres, il ne sera pas en mesure d'approcher efficacement la fonction de décision. Il s'agit ainsi de trouver des modèles capables dont le nombre de degré de liberté permet d'exploiter au mieux l'ensemble des exemples d'apprentissage.

L'émergence de l'apprentissage profond dans les années 2000 a fortement contribué à renouveler la littérature en apprentissage statistique. En particulier, les réseaux de neurones profonds, bien que théorisés et mis en application dès les années 1960, ont trouvé une résonance particulière avec l'ère des données massives. Les grandes bases de données d'apprentissage, combinées avec des modèles de réseaux de neurones artificiels profonds et des implémentations parallèles permettant de rendre les calculs traitables en temps raisonnable, ont permis d'importantes avancées en perception artificielle. Le traitement d'images a largement bénéficié de cette conjonction. Dès 2012, les réseaux de neurones convolutifs profonds se sont imposés comme le nouvel état de l'art pour la classification d'images et ont petit à petit conquis une grande partie des tâches de perception visuelle.

1.2.3 Vision par ordinateur

La vision par ordinateur regroupe l'ensemble des techniques conçues pour l'interprétation automatisée des images. Dès les années 60, les experts en intelligence artificielle se sont penchés sur la possibilité de simuler les capacités sensorielles humaines. La vision étant le sens humain le plus mis à contribution, il est naturel que les travaux en ce sens aient été nombreux. La démocratisation des appareils photos et caméras numériques a d'autant plus accéléré les possibilités offertes en traitement d'images, non seulement grâce aux traitements correcteurs embarqués au sein-même des capteurs, mais également par la facilité nouvelle du post-traitement sur ordinateur.

Le Graal de la vision par ordinateur consiste à émuler la capacité du cerveau humain à interpréter une scène dynamique à partir des signaux visuels, en particulier à identifier rapidement les objets et à anticiper leurs mouvements. Ces fonctions cognitives sont indispensables pour la navigation autonome en robotique, mais bénéficient également aux applications en fouille de données. La numérisation automatique de documents anciens, la recherche en ligne d'images similaires ou l'audio-description automatique sont autant d'exemples de tâches pouvant s'appuyer sur une brique d'interprétation d'images.

En particulier, les efforts de la communauté de la vision par ordinateur se sont concentrés sur la reconnaissance d'objets dans des images, incluant à la fois leur identification et leur localisation. De nombreux descripteurs *ad hoc* ont été introduits pour des applications aussi variées que la détection de visages, la classification automatique de photographies d'espèces animales ou la reconnaissance optique de caractères. Le dénominateur commun de ces travaux est de chercher à donner du sens aux images. Extraire la sémantique d'une information visuelle non structurée est également l'objectif de l'observation de la Terre.

Cette thèse se trouve ainsi au croisement entre la télédétection, la vision par ordinateur et l'apprentissage automatique. En particulier, nous nous proposons de mettre en œuvre des méthodes d'apprentissage profond pour l'interprétation automatique d'images d'observation de la Terre.

1.3 Problématique

L'objectif de cette thèse est de proposer des méthodes d'apprentissage profond permettant de cartographier automatiquement la Terre en tirant profit des grandes quantités d'images acquises chaque jour. En particulier, il s'agit de sémantiser les images sous formes de cartes thématiques, notamment pour l'occupation des sols. Plusieurs questions découlent de cet objectif :

— Quels outils mettre en œuvre pour la cartographie automatique à partir d'images d'observation de la Terre ?
— Comment exploiter les multiples capteurs multispectraux, hyperspectraux et Lidar dans un cadre d'apprentissage profond ?
— Peut-on mettre à profit la revisite d'une même zone par plusieurs instruments pour enrichir les informations géographiques ?

— À quel point les modèles statistiques supervisés peuvent-ils s'appliquer à large échelle, pour cartographier l'intégralité de la planète ?
— Est-il possible d'extraire des images une information spatialement structurée pouvant renseigner sur la nature des objets et leur agencement géographique ?

En premier lieu, les outils pour l'interprétation automatique d'images sont légion. Si les réseaux de neurones artificiels sont populaires dans l'état de l'art pour la vision par ordinateur, leurs succès sont récents et leur introduction pour la télédétection est encore nouvelle. Il sera donc nécessaire dans un premier temps d'étudier le comportement des réseaux convolutifs profonds par rapport aux approches de l'état de l'art en classification d'images de télédétection. Le Chapitre 2 est l'occasion de rappeler les fondamentaux théoriques de l'apprentissage profond, et plus particulièrement des réseaux de neurones convolutifs ainsi que leurs applications en perception artificielle. Le Chapitre 3 démontre les limites du paradigme de classification par région dans le cadre de la cartographie automatisée d'images aériennes et met en avant la pertinence des réseaux entièrement convolutifs sur cette tâche.

Cependant, comme nous l'avons vu précédemment, les dispositifs d'observation de la Terre couvrent des domaines de longueur d'onde bien différents des appareils photographiques habituels. En outre, les capteurs optiques sont parfois complétés par des appareils Lidar qu'il est nécessaire de savoir exploiter pour interpréter les images de télédétection le plus fidèlement possible. En effet, l'observation de la Terre passe couramment par l'acquisition de données complémentaires sur une même zone à partir de capteurs hétérogènes. Qui plus est, les nouvelles bases de données géographiques libres d'accès représentent une source de données inédite encore inexploitée. La possibilité de fusionner ces données afin de tirer profit des avantages de chaque capteur serait donc un atout majeur en cartographie. Par conséquent, le Chapitre 4 étendra ainsi les résultats obtenus sur des images rouge-vert-bleu (RVB) à l'ensemble des données issues d'appareils multispectraux et hyperspectraux, ainsi qu'aux modèles de terrains dérivés des acquisitions *Light Detection And Ranging* (Lidar). Le Chapitre 5 introduit par la suite différentes stratégies d'apprentissage multimodal pour les réseaux profonds afin de répondre à la problématique de fusion de données, aussi bien dans le cas multicapteur que pour la prise en compte de connaissances *a priori*.

La généralisation des modèles statistiques étant par ailleurs l'élément fondamental permettant de passer à l'échelle, il est nécessaire d'étudier la capacité de généralisation des réseaux profonds sur de grands jeux de données comportant une large variété de scènes. En effet, cartographier l'intégralité de la surface du globe nécessite une robustesse du modèle aux variations locales de la biosphère, qu'elles soient spatiales ou temporelles. Notamment, l'apprentissage supervisé n'est parfois possible que sur des jeux de données de petite taille, à partir desquels il n'est pas trivial de construire des modèles généralistes. Le Chapitre 6 s'intéresse à ces deux problèmes.

Enfin, si l'interprétation d'images de télédétection permet de construire des cartes thématiques, c'est bien souvent les relations entre les sous-parties de ces cartes qui sont intéressantes. En particulier, l'analyse de niveau objet est une approche incontournable en géographie, car elle seule permet de modéliser les structures et leur agencement. Le Chapitre 7 explore donc différentes possibilités pour donner une structure spatiale aux cartes produites par les modèles de classification dense pixel à pixel proposés jusqu'ici.

Le Chapitre 8 clôt ce manuscrit et discute des pistes de recherches futures envisageables à la suite de ces travaux.

1.4 Contributions

Cette thèse apporte 4 contributions majeures.

1. Elle participe à établir les réseaux de neurones convolutifs comme nouvel état de l'art pour la cartographie automatisée d'images de télédétection.

2. Elle démontre la possibilité d'étendre les domaines d'application desdits réseaux à l'ensemble des capteurs optiques usuels.
3. Elle montre qu'il est envisageable et pertinent d'utiliser les informations présentes dans plusieurs capteurs lorsque cela est possible.
4. Elle montre que ces approches ne se confinent pas à des cas spécifiques, mais peuvent s'étendre à l'ensemble du globe.

Ces travaux ont fait l'objet de plusieurs publications :

Publications en revues internationales à comité de lecture

Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « Segment-before-Detect : Vehicle Detection and Classification through Semantic Segmentation of Aerial Images ». Dans : *Remote Sensing* 9.4 (13 avr. 2017), p. 368. DOI : 10.3390/rs9040368

Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « Beyond RGB : Very High Resolution Urban Remote Sensing with Multimodal Deep Networks ». Dans : *ISPRS Journal of Photogrammetry and Remote Sensing* (23 nov. 2017). ISSN : 0924-2716. DOI : 10.1016/j.isprsjprs.2017.11.011

Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « Deep Learning for Classification of Hyperspectral Data : A Comparative Review ». Dans : *IEEE Geoscience and Remote Sensing Magazine* in press (mar. 2019)

Publications en conférences internationales à comité de lecture

Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « How Useful Is Region-Based Classification of Remote Sensing Images in a Deep Learning Framework ? » Dans : *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. Juil. 2016, p. 5091-5094. DOI : 10.1109/IGARSS.2016.7730327

Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « Semantic Segmentation of Earth Observation Data Using Multimodal and Multi-Scale Deep Networks ». Dans : *Computer Vision – ACCV 2016*. Springer, Cham, 20 nov. 2016, p. 180-196. DOI : 10.1007/978-3-319-54181-5_12

Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « Fusion of Heterogeneous Data in Convolutional Networks for Urban Semantic Labeling ». Dans : *2017 Joint Urban Remote Sensing Event (JURSE)*. Mar. 2017, p. 1-4. DOI : 10.1109/JURSE.2017.7924566

Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « Joint Learning from Earth Observation and OpenStreetMap Data to Get Faster Better Semantic Maps ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Honolulu, United States, juil. 2017, p. 1552-1560. DOI : 10.1109/CVPRW.2017.199

Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « Generative Adversarial Networks for Realistic Synthesis of Hyperspectral Samples ». Dans : *2018 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. Juil. 2018, p. 5091-5094



Chapitre

2 État de l'art

An algorithm must be seen to be believed.

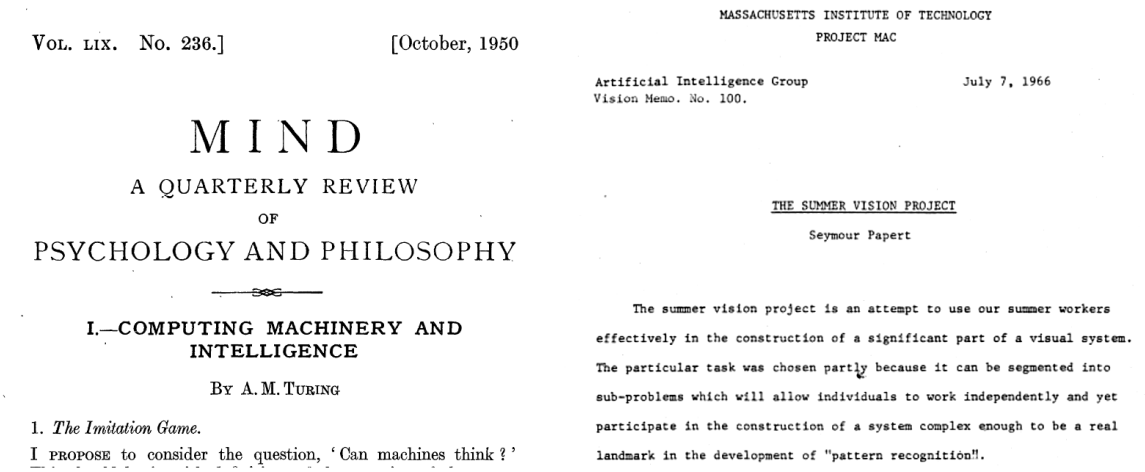
— Donald Knuth

Sommaire

2.1 Apprentissage profond pour la vision artificielle 10
2.1.1 Historique de l'apprentissage profond 10
2.1.2 Réseaux de neurones artificiels 13
2.1.3 Entraînement des réseaux de neurones 16
2.1.4 Réseaux de neurones convolutifs profonds 21
2.2 Apprentissage profond pour la segmentation sémantique 28
2.2.1 De la classification à la segmentation 28
2.2.2 Approches entièrement convolutives 35
2.3 Apprentissage pour le traitement d'images de télédétection 37
2.3.1 Différents types d'imagerie 37
2.3.2 Apprentissage et images de télédétection 39

Résumé du chapitre :

Ce chapitre introduit les bases théoriques en apprentissage profond sur lesquelles s'appuie la suite du manuscrit. Dans un premier temps, nous rappellerons brièvement les motivations et les grandes étapes de construction des réseaux de neurones artificiels modernes avant de détailler le fonctionnement des réseaux convolutifs profonds. Dans un second temps, nous étudierons plus en détail les applications de ces réseaux en vision par ordinateur et notamment pour la segmentation sémantique. Enfin, nous présenterons un tour d'horizon des méthodes classiques d'interprétation d'images de télédétection en nous concentrant sur les approches d'apprentissage et les spécificités liées à l'observation de la Terre.



Vol. lix. No. 236.] [October, 1950

MIND

A QUARTERLY REVIEW

OF

PSYCHOLOGY AND PHILOSOPHY

I.—COMPUTING MACHINERY AND INTELLIGENCE

By A. M. Turing

1. *The Imitation Game.*

I propose to consider the question, 'Can machines think?'

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
PROJECT MAC

Artificial Intelligence Group
Vision Memo. No. 100.

July 7, 1966

THE SUMMER VISION PROJECT

Seymour Papert

The summer vision project is an attempt to use our summer workers effectively in the construction of a significant part of a visual system. The particular task was chosen partly because it can be segmented into sub-problems which will allow individuals to work independently and yet participate in the construction of a system complex enough to be a real landmark in the development of "pattern recognition".

Figure 2.1 – Introductions de Turing [168] et Papert [132], deux documents fondateurs de l'intelligence artificielle et de la vision par ordinateur.

Cette thèse repose sur de nombreux travaux antérieurs en vision par ordinateur, intelligence artificielle et observation de la Terre. Dans ce chapitre, nous allons mettre en place le cadre scientifique fondamental sur lequel nous nous appuierons par la suite. Dans un premier temps, nous allons exposer les principes de l'apprentissage profond et formaliserons la théorie des réseaux de neurones artificiels convolutifs. Dans un second temps, nous détaillerons les applications de ces modèles d'apprentissage pour l'interprétation d'images, sous la forme d'une tâche dite de segmentation sémantique. Enfin, nous discuterons des études récentes sur l'interprétation automatisée d'images de télédétection, ce qui nous permettra d'identifier les spécificités applicatives liées à l'observation de la Terre.

2.1 Apprentissage profond pour la vision artificielle

2.1.1 Historique de l'apprentissage profond

La possibilité d'attribuer des capacités cognitives à un ordinateur est formalisée pour la première fois par Alan Turing en 1950 [168] (cf. Figure 2.1). Dans « *Computing Machinery and Intelligence* », Turing s'intéresse au cadre formel qui permettrait de répondre à la question suivante : « une machine peut-elle penser ? ». Selon lui, un ordinateur intelligent serait défini par sa capacité à imiter un humain, de manière à ce que d'autres individus soient incapables de discerner sa véritable nature. Toutefois, Turing ne répond pas à la question du « comment » et ne propose pas de solution à ce défi.

Pourtant, en 1943, Warren McCulloch et Walter Pitts proposaient déjà un système de neurones artificiels booléens [113] dotés de deux états : actif ou inactif. Ils définissent formellement un neurone comme étant un automate fini muni d'une fonction de transfert, permettant de transformer un ensemble d'entrées en une valeur de sortie. Certains neurones ne recevaient aucun signal d'un autre neurone, mais constituaient eux-mêmes le signal d'entrée. Les autres neurones calculaient alors des combinaisons logiques à partir des signaux qu'ils recevaient. McCulloch et Pitts [113] montrent que de nombreux prédicats de logique temporelle sont calculables par de tels réseaux booléens. En particulier, une extension de cette théorie réalisée par Stephen Kleene s'intéresse aux réseaux booléens dont le graphe présente des cycles, c'est-à-dire des réseaux *récurrents*. Kleene [82] prouve notamment que ces réseaux, qui constituent en réalité des automates finis, sont capables de modéliser n'importe quel langage rationnel [1].

En parallèle, le neuropsychologue Donald Hebb étudie les mécanismes cognitifs d'apprentissage au sein du cerveau. Il théorise ainsi l'apprentissage hebbien, un principe selon lequel

1. C'est-à-dire un langage défini par une expression régulière.

la connexion entre deux neurones se renforce à chacune de leurs activations simultanées [65]. De plus, Hebb suggère que certains neurones se regroupent en « assemblées de cellules » qui s'activent de façon synchronisée, codant ainsi une représentation mentale des signaux envoyés au cerveau. Comme nous allons le voir, ces deux idées constituent des sources d'inspiration considérable pour les stratégies d'apprentissage développées pour l'intelligence artificielle.

En 1957, Frank Rosenblatt définit le perceptron [142]. Il s'agit d'un réseau de neurones acyclique, comme ceux de McCulloch et Pitts [113]. Les entrées et sorties sont booléennes et le réseau ne possède qu'une unique couche. Les poids des connexions sont néanmoins déterminées automatiquement, en utilisant la règle de Hebb [65]. À la même époque, Bernard Widrow fabrique l'*Adaptive Linear Neuron* (ADALINE) [180], une machine à base de memistors s'inspirant elle aussi du modèle de McCulloch et Pitts. L'ADALINE est très proche du perceptron dans sa conception : il s'agit d'un réseau linéaire à une couche opérant sur la somme pondérée de ses entrées. Pour déterminer les poids des connexions, Widrow adopte un algorithme de descente de gradient minimisant l'erreur quadratique du modèle. Cependant, ces deux modèles présentent une lacune majeure. En effet, le perceptron est assuré de pouvoir trouver une frontière séparant les données de manière optimale, mais uniquement si celles-ci sont linéairement séparables. Ces classifieurs étant linéaires, ils ne peuvent pas résoudre de problèmes non-linéairement séparables. Dans le livre *Perceptrons*, Minsky et Papert [116] prouvent qu'un perceptron à une seule couche cachée est incapable de reproduire la fonction XOR quel que soit le nombre de neurones utilisés, et ce en dépit de la simplicité apparente de l'opération. Malheureusement, il n'existe à l'époque aucune stratégie satisfaisante permettant de déterminer les poids optimaux d'un perceptron à plusieurs couches qui présenterait un caractère non-linéaire. Les réseaux de neurones artificiels tombent alors en désuétude pendant plusieurs années. Dans sa thèse soutenue en 1975 [179], Paul Werbos formalise un algorithme de descente de gradient pour la minimisation d'erreur dans un réseau de neurones à plusieurs couches en utilisant le théorème de dérivation des fonctions composées. Il nomme cet algorithme *backpropagation* (rétro-propagation). Il faudra toutefois attendre dix ans pour voir apparaître les premières implémentations de l'algorithme de rétro-propagation du gradient pour entraîner des perceptrons multicouche [144, 91]. [2]

L'étude théorique des réseaux de neurones à propagation avant, et notamment des perceptrons, reprend alors. En 1989, Cybenko [34] démontre le théorème d'approximation universelle prouvant que les fonctions calculables par un perceptron sont denses dans l'ensemble des fonctions continues par morceaux, dans le cas de la sigmoïde comme fonction de transfert. Hornik [69] généralise ce résultat deux ans plus tard à l'ensemble des fonctions d'activation usuelles. Le théorème formel est rappelé ci-dessous.

Théorème 1. *Soit φ une fonction bornée, croissante non-constante. Soit C_0^n l'ensemble des fonctions continues définies sur $[0,1]^n$. Alors :*

$$\forall \epsilon > 0, \forall F \in C_0^n, \exists N \in \mathbb{N}^*, \text{ des réels } v_i, b_i \in \mathbb{R} \text{ et des vecteurs } \mathbf{w}_i \in \mathbb{R}^n \text{ avec } i \in [\![1, n]\!] \text{ tels que}$$

$$\hat{F} : \mathbf{x} \to \sum_{i=1}^{N} v_i \varphi\left(\mathbf{w}_i^t \mathbf{x} + b_i\right)$$

soit une approximation de F à ϵ près, c'est-à-dire :

$$\forall \mathbf{x} \in [0,1]^n, \ \left|F(\mathbf{x}) - \hat{F}(\mathbf{x})\right| < \epsilon \, .$$

Ce résultat signifie que que toute fonction relativement régulière (continue par morceaux sur un ensemble de compacts) peut être approchée avec une précision arbitraire par un

2. L'ADALINE sera également adapté en multicouche avec sa variante MADALINE [181], utilisant un algorithme spécifique car utilisant la fonction d'activation signe dont le gradient est nul presque partout. Widrow et Lehr convergent deux ans plus tard vers une structure de Madaline utilisant la fonction sigmoïde comme activation, entraînable par rétropropagation.

perceptron. Il est ainsi prouvé que de tels réseaux de neurones artificiels peuvent simuler presque n'importe quelle fonction, sans pour autant disposer de méthode de construction de tels réseaux. Outre la théorie des perceptrons, leurs applications pratiques dans le cadre de la vision artificielle pour la reconnaissance de formes ont également été étudiées. La reconnaissance de caractères écrits, notamment les chiffres et les lettres, est un problème particulièrement populaire. En 1980, Fukushima [52] introduit *Neocognitron*, un perceptron multicouche dont la structure s'inspire des travaux de Hubel et Wiesel [75, 76] sur les cortex visuels des chats et des singes. Le *Neocognitron* extrait de l'image des caractéristiques locales robustes aux légères déformations, qui sont graduellement combinées en cascade par le réseau. De cette façon, le modèle n'est plus seulement sensible à l'intensité des pixels, mais aux motifs présents dans les variations locales, se rapprochant ainsi du fonctionnement des yeux animaux [96]. En 1989, LeCun et al. [92] proposent une architecture de perceptron multicouche pour la reconnaissance de chiffres manuscrits dont la première couche est *convolutive*, entraîné par rétropropagation. Sur ce principe, ils élaborent ensuite l'architecture LeNet-5 [94], premier réseau de neurones convolutif, ou *Convolutional Neural Network* (CNN) moderne. En 2004, les méthodes de détection et classification d'objets par réseau de neurones convolutif, ou *Convolutional Neural Network* (CNN) sont évaluées comme étant compétitives, voire supérieure, par rapport aux machine à vecteurs de support, ou *Support Vector Machine* (SVM), opérant directement sur les pixels. Les premiers travaux utilisant les représentations apprises par les CNN pour remplacer les descripteurs images *ad hoc* comme SIFT *(Scale-Invariant Feature Transform)* [107] ou les histogrammes de gradient orientés (HOG) [35] pour la classification d'objets apparaissent dans les années 2000 [151, 72].

En 2006, Hinton et Salakhutdinov [67] introduisent les réseaux de neurones auto-encodeurs capables de compresser un ensemble de données en les projetant dans un espace de dimension plus faible. Leur approche pour la réduction de dimension utilise une pile de *Restricted Boltzmann Machines* (RBM) [1, 146], entraînées successivement couche par couche. Ce modèle hybride est exploré dans un article de 2006 [66] qui leur donne le nom de *Deep Belief Networks* (DBN). L'année suivante, Bengio et al. [10] étendent ce préentraînement par couche à des DBN pour la régression. Leurs travaux suggèrent notamment que le préentraînement permet d'initialiser les couches supérieures à partir de meilleures représentations des abstractions de haut niveau que le hasard. Yoshua Bengio défend par ailleurs l'idée qu'un bon algorithme d'apprentissage doit être capable d'apprendre des représentations sémantiques pertinentes à des niveaux d'abstraction variés à partir de données non nécessairement annotées, c'est-à-dire en apprentissage non supervisé [8]. Il argumente en faveur des modèles profonds comme étant plus expressifs grâce à leur capacité à apprendre des représentations à partir des données, en s'appuyant sur les progrès récents en sciences cognitives concernant le cortex visuel [152]. L'introduction de fonctions d'activation non-saturantes comme la *Rectified Linear Unit* (ReLU) [56] permet de s'affranchir du préentraînement et des problèmes d'explosion des gradients, qui rendaient jusqu'alors impossible l'entraînement de réseaux très profonds.

C'est également en 2006 que les premières implémentations des CNN sur *Graphics Processing Unit* (GPU) voient le jour [22] pour le traitement automatisé de documents, puis pour l'apprentissage non-supervisé de DBN [140] et la reconnaissance de caractères [27]. En 2011, Dan Cireşan propose des méthodes à base de CNN qui obtiennent la première place dans deux compétitions : reconnaissance de caractères chinois [100] et classification de panneaux de signalisation [161]. Ces CNN obtiennent en outre d'excellents résultats en reconnaissances de caractères latins et en classification d'objets pour des petites images sur la base de données CIFAR-10 [26]. En 2010 débute la compétition de reconnaissance d'objets *ImageNet Large-Scale Visual Recognition Challenge* (ILSVRC), qui utilise comme référence la banque de données ImageNet [40]. Un million d'images sont annotées pour mille classes d'intérêt différentes. En 2012, la compétition est remportée par Krizhevsky, Sutskever et Hinton [84] à l'aide du réseau convolutif profond AlexNet implémenté sur GPU à l'aide de

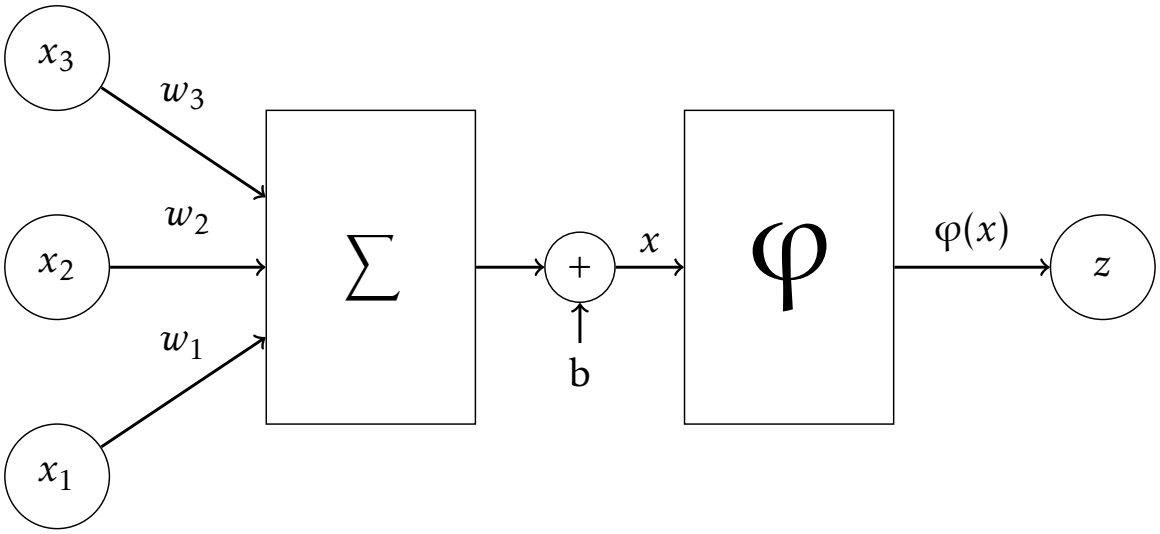


Figure 2.2 – Modélisation d'un neurone artificiel.

la bibliothèque *Compute Unified Device Architecture* (CUDA). AlexNet obtient 15% d'erreur *top-5* [3], tandis que la seconde méthode du podium n'obtient que 26%. Ce succès inattendu est à l'origine de l'explosion en popularité des réseaux convolutifs et de l'apprentissage profond dans la communauté de la vision par ordinateur. La compétition ILSVRC a été depuis remportée chaque année par les approches CNN, aussi bien pour la reconnaissance et la localisation que la segmentation d'objets. Le succès des réseaux convolutifs profonds depuis 2012 est donc dû à la convergence de trois facteurs : des avancées théoriques (ReLU, réseaux convolutifs) permettant d'entraîner des réseaux plus profonds, la mise à disposition de grandes bases de données annotées pour l'apprentissage supervisé et des implémentations efficaces sur GPU rendant les temps de calcul acceptables.

Dans la suite de cette partie, nous formalisons le cadre théorique des réseaux de neurones artificiels et aux méthodes permettant leur entraînement pour différentes tâches avant de s'intéresser plus particulièrement aux modèles convolutifs.

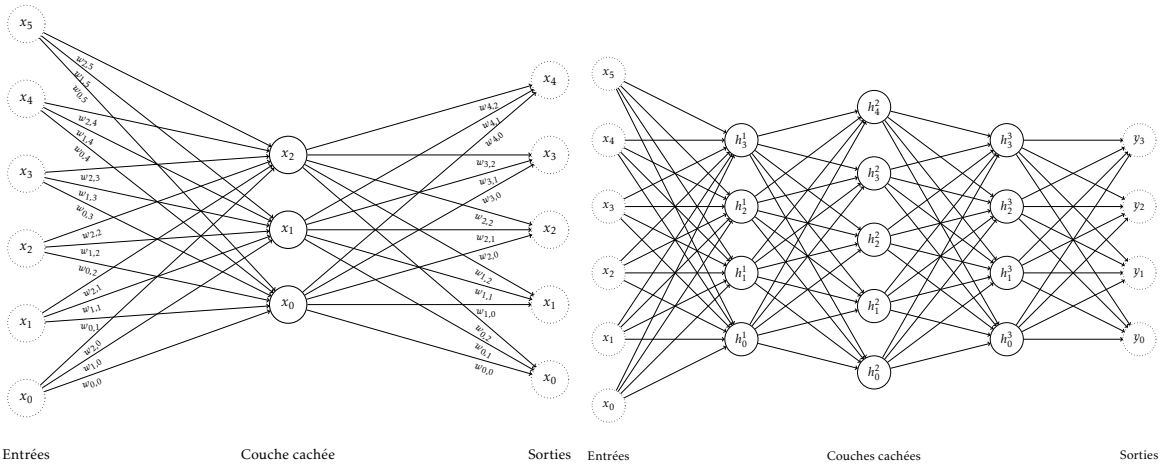
2.1.2 Réseaux de neurones artificiels

La définition formelle d'un neurone artificiel a été introduite par McCulloch et Pitts [96] en 1959. Un neurone doté d'une fonction de transfert φ opère sur un ensemble de n neurones d'entrée émettant chacun une valeur $x_1 \dots x_n$, auxquels il est connecté par des synapses de poids w_i. La valeur d'entrée x du neurone correspond à la somme des signaux d'entrée pondérés par les poids de leur synapse. Le neurone émet en sortie l'image $z = \varphi(x)$ de ce signal par sa fonction de transfert. Le schéma de la Figure 2.2 décrit ce procédé. L'activation en sortie d'un neurone s'obtient donc par la formule

$$z = \varphi\left(\sum_{i=1}^{n} w_i x_i + b\right). \tag{2.1}$$

Plusieurs neurones peuvent être connectés les uns aux autres et forment alors un graphe orienté et pondéré. Un réseau de neurones à propagation avant désigne un graphe neuronal acyclique. En pratique, on considère des graphes k-partis que l'on peut alors représenter par « couches ». Dans le cas d'un perceptron à couches multiples, les valeurs des signaux d'entrée et de sortie sont placées dans des couches spécifiques. Les couches de neurones réellement optimisables sont nommées « couches cachées » et font l'interface entre l'entrée et

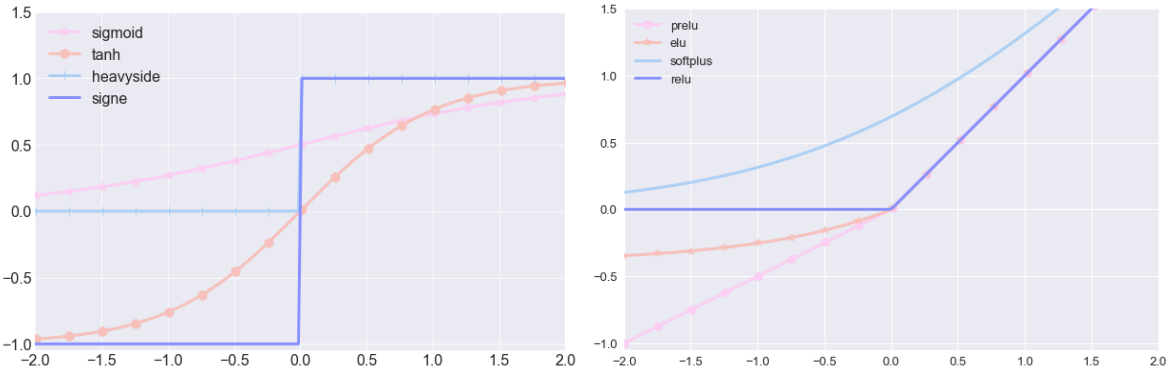
3. Une prédiction *top-5* est considérée juste si la classe recherchée fait partie des cinq premières données par le modèle.



(a) Perceptron à une couche cachée.

(b) Perceptron multicouche.

FIGURE 2.3 – Perceptron à une et plusieurs couches. Les entrées et sorties peuvent être de dimensions variables et sont représentées comme des neurones.



(a) Fonctions d'activation saturantes.

(b) Fonctions d'activation non-saturantes.

FIGURE 2.4 – Exemples de fonctions d'activation.

la sortie. Ces réseaux possèdent ainsi une topologie fixée au préalable et sont paramétrés par l'ensemble des poids affectés aux connexions entre neurones. Les perceptrons multicouches à une et plusieurs couches cachées sont illustrés dans la Figure 2.3. Ce type de couche dont l'ensemble des neurones sont reliés à ceux de la couche suivante est appelé « couche entièrement connectée » et est une des briques fondamentales des réseaux profonds modernes.

La fonction d'activation des neurones peut prendre de nombreuses formes. Il est *a minima* nécessaire que celle-ci soit non-linéaire, sans quoi l'expressivité du réseau se trouve limitée à celle du perceptron, et presque partout différentiable, afin de pouvoir appliquer l'algorithme de rétro-propagation du gradient. La fonction d'activation φ est en outre généralement choisie telle que φ et sa dérivée φ' soient monotones croissantes. La Figure 2.4a illustre plusieurs activations communément utilisées dans les réseaux de neurones artificiels profonds :

— La **sigmoïde**, ou fonction logistique : $\sigma(x) = \frac{1}{1+e^{-x}}$.
— La fonction **tangente hyperbolique** : $\tanh(x) = \frac{1-e^{-2x}}{1+e^{-2x}}$.
— La **marche de Heavyside** : $H(x) = 0$ si $x < 0$ et 1 si $x \geq 0$.
— La fonction **signe** : $\text{signe}(x) = +1$ si $x > 0 et - 1$ si $x < 0$.

La marche de Heavyside, tout comme la fonction signe, a des gradients nuls presque partout, sa dérivée étant l'impulsion de Dirac δ. Ceci les rend peu utilisées en pratique car il est impossible de leur appliquer l'algorithme de rétro-propagation du gradient. La fonction sigmoïde, bien qu'initialement la plus commune, est une fonction saturante qui souffre de gradients évanescents. Ce problème n'est pas circonscrit à la sigmoïde et est particulièrement

pregnant dans le cas des réseaux de neurones récurrents [68]. L'accumulation de couches dans le réseau rend géométrique l'évolution des amplitudes du gradient durant la rétro-propagation du gradient. Le produit cumulé de n gradients dans des fonctions d'activation à tendance contractante produit un $n+1^{e}$ gradient plus faible que le précédent, et ainsi de suite. À l'inverse, il est possible d'avoir des gradients explosifs dont la norme croît exponentiellement. Ce problème est empiré dans le cas des fonctions saturantes comme la sigmoïde ou la tangente hyperbolique car leurs gradients appartiennent nécessairement à l'intervalle $[0,1]$. L'utilisation de la sigmoïde ou la tangente hyperbolique a varié dans la littérature. LeCun et al. [93] recommande d'utiliser la fonction hyperbolique modifiée $f(x) = 1{,}7159 tanh(\frac{2}{3}x)$, notamment car celle-ci est bornée par $[-1,1]$ et centrée en 0, ce qui est adapté au travail sur des données normalisées à moyenne nulle et variance unitaire.

Désormais, les fonctions d'activation non-linéaires non-saturantes sont majoritairement utilisées dans la littérature afin de limiter les gradients évanescents. En effet, Glorot, Bordes et Bengio [56] ont proposé d'utiliser la fonction ReLU, introduite précédemment pour les DBN [119], et la fonction SoftPlus pour n'importe quel type de réseau de neurones. Ils analysent les effets de ces fonctions d'activation sur l'optimisation des réseaux et aboutissent à trois conclusions. Tout d'abord, les fonctions d'activation non-linéaires généralisent dans l'ensemble mieux que les réseaux utilisant *tanh*. Ensuite, les réseaux entraînés avec ReLU ne nécessitent pas de préapprentissage non-supervisé couche par couche, ce qui accélère grandement les temps d'entraînement. Enfin, ces modèles sont plus parcimonieux que leurs équivalents usuels. Compte-tenu de la simplicité d'implémentation des ReLU et leur efficacité calculatoire, ces fonctions ont rapidement été adoptées par la communauté.

Dans l'ensemble, bien que ces hypothèses ne semblent pas nécessaires à la construction de réseaux profonds [126], la plupart des fonctions d'activation couramment utilisées sont continues, monotones, contractantes et s'appliquent indépendamment sur chaque activation. Plusieurs variantes ont été proposées autour des fonctions linéaires rectifiées, notamment une version paramétrique dont la partie négative est de pente $\alpha > 0$, fixée par l'utilisateur (*Leaky ReLU* [109]) ou apprenable (*Parametrized Rectified Linear Unit* (PReLU) [63]). Une alternative dérivable partout a également été proposée sous la forme de *Exponential Linear Unit* (ELU) [29]. Certaines de ces variantes sont illustrées dans la Figure 2.4b et leur formule est donnée ci-dessous :

- La fonction **ReLU** : $\mathrm{ReLU}(x) = \max(0, x)$.
- La fonction **SoftPlus** : $s^{+}(x) = \ln(1 + e^{x})$.
- La fonction ***Leaky ReLU*** : $\mathrm{LReLU}_{\alpha}(x) = max(0, x) - \alpha max(0, -x)$, avec α un hyperparamètre.
- La fonction **PReLU** : $\mathrm{PReLU}(x, \alpha) = max(0, x) - \alpha max(0, -x)$, avec α optimisable.
- La fonction **ELU** : $\mathrm{ELU}_{\alpha}(x) = x$ si $x > 0$ et $\alpha(exp(x) - 1)$ sinon.

Le théorème d'approximation universelle [34, 69] indique que l'ensemble des fonctions engendrées par les perceptrons est dense dans l'ensemble des fonctions continues par morceaux sur des compacts. Autrement dit, toute fonction $f : \mathrm{E} \to \mathbb{R}^{m}$ avec $\mathrm{E} = \bigcup_{k} \mathrm{C}_{k}$ une union de compacts de $\mathbb{R}^{n}$ continue sur chaque compact peut être approchée à une précision $\epsilon > 0$ arbitraire par un perceptron. Cependant, deux limites viennent borner ce résultat. Tout d'abord, le théorème généralisé par Hornik ne considère que la classe des fonctions d'activation monotones non-constantes et bornées, ce qui exclut les fonctions linéaires rectifiées comme ReLU. Sonoda et Murata [159] ont néanmoins levé cette limitation en montrant que des réseaux dotés de fonctions d'activation non-bornées satisfont tout de même le théorème d'approximation universelle.

La deuxième limitation est plus fondamentale. Le théorème représente une garantie théorique de l'existence d'un ensemble de paramètres permettant d'approcher la fonction souhaitée, mais ne donne ni de bornes sur le nombre de neurones nécessaires, ni de méthode de construction. Ainsi, il n'y aucune certitude que les poids puissent être atteignables par descente de gradient, et nous ne disposons d'aucune indication sur la topologie du réseau.

En particulier, le théorème vaut pour des réseaux superficiels à une seule couche, tandis que le consensus scientifique tend à préférer des réseaux de plus en plus profonds. Ceux-ci permettent notamment d'approcher des fonctions plus complexes en utilisant moins de neurones [11, 115]. La structure hiérarchique des réseaux multicouches semble particulièrement adaptée pour simuler des fonctions composées en contournant la malédiction de la dimension [137]. Toutefois, cela augmente également le nombre d'hyperparamètres à régler pour définir l'architecture du réseau. L'absence de méthode systématique de construction des architectures de réseau profond conduit donc à l'utilisation intensive de l'approche essai-erreur ou à des méthodes de méta-apprentissage [194].

L'étude des propriétés théoriques des réseaux profonds reste un problème ouvert de recherche en mathématiques appliquées. L'expressivité des réseaux, c'est-à-dire leur capacité à représenter certaines classes de fonctions, semble augmenter avec leur profondeur plus rapidement qu'avec leur largeur [98]. Les introductions des nouvelles architectures et des fonctions d'activation non-saturantes nécessitent de mettre à jour les résultats d'approximation universelle et d'expressivité, la communauté s'attachant ainsi à comprendre les mécanismes mathématiques expliquant les succès empiriques des réseaux profonds modernes [108]. En plus d'ouvrir la « boîte noire », ces travaux sont indispensables afin d'établir des fondations solides pour l'apprentissage profond, afin que la conception des architectures neuronales repose à terme sur des résultats théoriques et non plus des intuitions.

2.1.3 Entraînement des réseaux de neurones

L'absence de méthode constructive pour déterminer les poids optimaux d'un réseau de neurones artificiels contraint à chercher des heuristiques d'optimisation. Ainsi, l'algorithme de rétro-propagation [179, 144, 91] utilise la descente de gradient pour estimer un jeu de paramètres adéquat. Bien que rien ne garantisse que les poids optimaux dont l'existence est assurée par le théorème d'approximation universelle ne soient atteignables de cette façon, cette méthode suffit en pratique.

L'algorithme de descente de gradient [20] est appliqué au modèle afin de minimiser l'erreur totale en mettant à jour les poids du réseau. Cet algorithme, dit de la plus forte pente, permet d'approcher un minimum local d'une fonction f différentiable en cherchant ses points stationnaires, c'est-à-dire les points où son gradient est nul. L'algorithme repose sur l'idée que f décroît le plus rapidement dans la direction opposée à son gradient et fonctionne de la façon suivante.

Définition 1. *Algorithme de descente de gradient :*

Soit $f : \mathbb{R}^n \to \mathbb{R}$ une fonction différentiable et ∇f son gradient. Soit $x_0 \in \mathbb{R}^n$ un point initial, $\epsilon > 0$ le seuil de tolérance de l'algorithme et $\alpha > 0$ le pas de la descente. On définit alors la suite $(x_i)_{i\geq 0} \in \mathbb{R}^{\mathbb{N}}$ telle que :

$$x_{i+1} = x_i - \alpha \nabla f(x_i) \, .$$

L'algorithme s'arrête lorsque $\nabla f(x_i) \leq \epsilon$ et renvoie x_i.

Dans le cas des réseaux de neurones, la fonction à minimiser est appelée « fonction objectif » ou « fonction de coût », notée $\mathcal{L}$ par la suite. Généralement, $\mathcal{L}$ est une indicatrice de l'erreur totale du modèle sur l'ensemble du jeu de données d'entraînement Ω, que l'on va chercher à minimiser. Ainsi, l'optimisation du réseau de neurones consiste à résoudre l'équation :

$$W^* = \text{argmin}_W \, \mathcal{L}(W, \Omega) \tag{2.2}$$

pour un modèle paramétré par les poids de ses connexions $W = \{w_1, \ldots, w_m\}$ et de fonction objectif $\mathcal{L}$.

Tant que la fonction de coût $\mathcal{L}$ est dérivable, il est possible de la minimiser en utilisant l'algorithme de descente de gradient. En particulier, on calcule alors la mise à jour des poids du modèle en rétro-propageant la valeur du gradient d'une couche à la précédente : c'est

l'algorithme de rétro-propagation [179, 93, 144]. La mise à jour des poids du réseau se fait ainsi dans la direction opposée du gradient par rapport à ces mêmes poids $\nabla_W \mathcal{L}(W, \Omega)$.

Définition 2. *Algorithme de descente de gradient appliqué à un réseau de neurones :*

1. *Initialiser aléatoirement les poids* W.
2. *Calculer* $\nabla_W \mathcal{L}(W, \Omega)$ *sur l'ensemble du jeu de données.*
3. *Tant que* $\nabla_W \mathcal{L}(W, \Omega) > \epsilon$ *:*
   — $W := W - \alpha \nabla_W \mathcal{L}(W, \Omega)$

En pratique, les bases de données d'apprentissage peuvent contenir des millions d'exemples et le cardinal de Ω devient alors très grand. Par conséquent, on applique généralement une variante en ligne de l'algorithme, appelée descente de gradient *stochastique*. Cette variante effectue la mise à jour des poids pour chaque exemple d'apprentissage, en estimant l'erreur moyenne à partir de l'erreur sur un seul échantillon.

Définition 3. *Algorithme de descente de gradient stochastique :*

1. *Initialiser aléatoirement les poids* W.
2. *Tant que le critère d'arrêt n'est pas atteint :*
   — *Tirer aléatoirement un exemple d'apprentissage* $\omega \in \Omega$
   — $W := W - \alpha \nabla_W \mathcal{L}(W, \omega)$

L'algorithme s'arrête lorsque le critère d'arrêt est vérifié, généralement lorsqu'un nombre d'itérations prédéfini est atteint.

L'estimation du gradient $\nabla_W \mathcal{L}(W, \omega)$ risque cependant d'être bruitée et peut ainsi subir des changements importants de direction entre deux itérations successives de l'algorithme. Afin de stabiliser la progression de l'algorithme, on utilise le plus souvent l'algorithme de descente de gradient stochastique *par mini-lots*, également appelés *mini-batches*. Le gradient global est alors estimé à partir d'une moyenne effectuée sur un mini-lot, ou *mini-batch*, de k échantillons :

Définition 4. *Algorithme de descente de gradient stochastique par mini-lots :*

1. *Initialiser aléatoirement les poids* W.
2. *Tant que le critère d'arrêt n'est pas atteint :*
   — *Tirer aléatoirement* k *exemples d'apprentissage* $(\omega_1, \ldots, \omega_k) \in \Omega^k$
   — $W := W - \alpha \frac{1}{k} \sum_{i=1}^{k} \nabla_W \mathcal{L}(W, \omega_i)$

L'algorithme s'arrête lorsque le critère d'arrêt est vérifié, généralement lorsqu'un nombre d'itérations prédéfini est atteint.

La mise à jour s'appliquant sur l'ensemble des couches du réseau, il est donc nécessaire de pouvoir calculer $\frac{\partial \mathcal{L}}{\partial w_i}$ pour chaque vecteur de poids w_i paramétrisant la i^e couche. Or, le calcul direct du gradient de $\mathcal{L}$ ne peut s'effectuer que sur la dernière couche. Pour remonter aux dérivées partielles des couches précédentes, il est nécessaire d'utiliser l'algorithme de rétro-propagation du gradient.

L'algorithme de rétro-propagation du gradient se fonde sur la règle de dérivation en chaîne, c'est-à-dire le théorème de dérivation des fonctions composées [38, 87] :

Théorème 2. *Soient* f *et* g *deux fonctions telles que* $f : I \to J \subset \mathbb{R}$ *et* $g : J \to \mathbb{R}$. *Soit* $x \in I$ *tel que* f *admet une dérivée en* x. *Alors, la fonction composée* $h = g \circ f : I \to \mathbb{R}$ *admet une dérivée en* x *de valeur :*

$$h'(x) = (g \circ f)'(x) = f'(x) \times g'(f(x)) .$$

Si f *et* g *sont dérivables respectivement sur* I *et* J, *alors :*

$$(g \circ f)' = f' \times (g' \circ f) .$$

ou encore, en utilisant la notation de Leibniz, avec $z = g(y)$ et $y = f(x)$:

$$\frac{\mathrm{d}z}{\mathrm{d}x} = \frac{\mathrm{d}z}{\mathrm{d}y} \times \frac{\mathrm{d}y}{\mathrm{d}x} \,.$$

Ce théorème s'étend aux dérivées partielles de fonctions à valeurs dans $\mathbb{R}^n$.

Pour diminuer l'erreur, il est nécessaire de mettre à jour les poids w^k dans la direction opposée au gradient $\frac{\mathrm{d}e}{\mathrm{d}w^k}$. En notant z^k les activations en sortie de la k^{e} couche, la dérivation en chaîne donne :

$$\frac{\partial \mathcal{L}}{\mathrm{d}w^k} = \frac{\partial \mathcal{L}}{\mathrm{d}z^k} \times \frac{\partial z^k}{\partial w^k} = \frac{\partial \mathcal{L}}{\partial z^{(k+1)}} \times \frac{\partial z^{(k+1)}}{\partial z^k} \times \frac{\partial z^k}{\partial w^k} \,.$$

Autrement dit, il est possible de remonter le réseau en partant des couches les plus profondes jusqu'aux premières couches afin de faire remonter le gradient $\frac{\partial \mathcal{L}}{\partial w^k}$. Pour obtenir le gradient de l'erreur par rapport aux poids d'une couche, il est nécessaire de calculer le gradient des sorties par rapport aux poids ainsi que le gradient des sorties par rapport aux entrées. Cette étape est appelée la propagation arrière (*backward pass*).

Dans la réalité, les fonctions mises en jeu opèrent non pas sur des vecteurs mais sur des tenseurs $\mathbf{x}, \mathbf{y}, \mathbf{z}$. Nonobstant, le théorème de dérivation des fonctions composées peut alors se réécrire en utilisant les jacobiennes $\mathbf{J}$:

$$\mathbf{J}_{\mathrm{F}\circ\mathrm{G}} = \mathbf{J}_{\mathrm{F}} \circ \mathrm{G} \cdot \mathrm{J}_{\mathrm{G}}$$

et l'algorithme de rétro-propagation s'applique encore.

C'est pour cette raison que les gradients peuvent devenir évanescents ou explosifs. La suite des normes des gradients successifs remontant le réseau est rendue quasi-géométrique par les multiplications successives. Lorsque la norme de la jacobienne est majoritairement inférieure à 1, l'amplitude des gradients tend vers 0. La convergence est alors lente, voire impossible. Si la norme est trop souvent supérieure à 1, alors les gradients croissent exponentiellement et les mises à jour des poids sont très instables. En pratique, on cherchera donc à obtenir des jacobiennes de norme unitaire, notamment lors de l'initialisation [148].

L'objectif du modèle est d'approcher une fonction $\mathcal{F}$. À cette fin, il est utile d'introduire une fonction de coût $\mathcal{L}$ mesurant l'erreur d'approximation commise par le réseau, de telle sorte que :

$$\mathcal{L}(\widehat{\mathcal{F}_{\mathrm{W}}}(x) - \mathcal{F}(x)) \to 0 \Rightarrow \hat{\mathcal{F}} \to \mathcal{F} \quad ,$$

c'est-à-dire que la minimisation de l'erreur implique la convergence du modèle vers la fonction réelle.

La nature de la fonction de coût varie en fonction de la tâche à réaliser. En régression, lorsque $\mathcal{F}$ est à valeurs continues, il est courant d'utiliser une distance sur l'espace des fonctions, comme les normes L_1 ou L_2. Pour chaque échantillon, on peut comparer la prédiction $\hat{y}$ avec la vérité terrain y par

$$\mathrm{L}_1(y, \hat{y}) = |\hat{y} - y|$$

$$\text{ou} \quad \mathrm{L}_2(\hat{y}, y) = \|\hat{y} - y\| \,.$$

Utiliser la norme L_2 revient à approcher $\mathcal{F}$ par la méthode des moindres carrés. En règle générale, la norme L_1 se montre plus robuste aux observations aberrantes, qui explosent dans le cas de la norme euclidienne. En comparaison, la norme L_2 a l'avantage d'être partout dérivable et sa tolérance aux faibles erreurs (fonction contractante sur $[-1, 1]$) la rend souvent plus robuste.

Lorsque $\mathcal{F}$ est à valeurs discrètes, notamment dans le cas d'une classification, y se présente sous la forme d'un encodage *one-hot*. Pour un problème à n classes, l'appartenance de y à la k^{e} classe se traduit sous forme d'un vecteur $y_i = \delta_{i,k}$ avec δ le delta de Kronecker. Autrement dit, y est encodé sous la forme $(0, \ldots, 0, 1, 0, \ldots, 0)$, c'est-à-dire un vecteur dont toutes les composantes sont nulles à l'exception de la k^{e}. Il est bien entendu possible d'utiliser

les mêmes fonctions de coût que précédemment, mais on préfère généralement minimiser la fonction d'entropie croisée. Celle-ci se calcule de la façon suivante :

$$\mathrm{H}(z, y) = -\sum_{i=1}^{n} y_i \log(z_i) \ . \tag{2.3}$$

L'entropie croisée est particulièrement intéressante en classification car sa minimisation coïncide avec celle de la divergence de Kullback-Leibler entre la distribution statistique des $\hat{y}$ et des y, c'est-à-dire l'image de $\mathcal{F}$ et celle de $\hat{\mathcal{F}}$. Pour que cela se vérifie, il est nécessaire que $\hat{y}$ soit un vecteur de probabilité tel que $\hat{y}_i \in [0, 1]$ et $\sum_i \hat{y}_i = 1$. Pour ce faire, les activations en sortie du réseau sont donc passées dans une fonction *softmax* :

$$\hat{y}_i = z_i = softmax(x)_i = \frac{\exp(x_i)}{\sum_j \exp(x_j)} \tag{2.4}$$

qui est une généralisation de la sigmoïde au cas multiclasse.

Soulignons que l'algorithme de descente de gradient ne dispose d'une convergence garantie que dans le cas où la fonction $\mathcal{L}$ est convexe, ce qui n'est jamais le cas en pratique pour des réseaux profonds. Plusieurs variantes de l'algorithme stochastique ont été proposées pour en améliorer les propriétés de convergence. Le pas de la descente, noté α dans la Définition 1, joue notamment un rôle important dans l'optimisation des réseaux de neurones profonds. Dénommé taux d'apprentissage, α contrôle l'amplitude des mises à jour des poids du modèle. Si α est trop élevé, les mises à jour à chaque itération seront importantes et la convergence instable. À l'inverse, un α faible ralentit la convergence et peut bloquer celle-ci dans des minima locaux. Les variantes de la descente de gradient stochastique introduisent des heuristiques spécifiques pour la mise à jour des poids. En particulier, les méthodes dites avec « moment » s'inspirent du moment cinétique en mécanique afin de conserver une partie de la vélocité du gradient entre chaque itération, afin de limiter les oscillations le long des courbes de niveaux de la surface d'erreur [139, 121]. Sutskever et al. [162] ont notamment montré que les méthodes avec moment permettent d'améliorer les performances finales du modèle, y compris dans le cas de poids initiaux mal choisis. Polyak et Juditsky [138] proposent par ailleurs une descente de gradient par mini-lot asynchrone en moyennant le gradient sur les n dernières itérations lors de la mise à jour des poids.

D'autres variantes introduisent des politiques d'ajustement du taux d'apprentissage α au cours de l'entraînement. En effet, rien ne contraint α à être constant dans l'algorithme de descente de gradient. Il est possible de l'ajuster manuellement lors de l'apprentissage, par exemple en commençant avec un taux d'apprentissage élevé au début, puis en le multipliant par une constante $\gamma < 1$ à intervalles réguliers. Bottou [16] recommande ainsi une descente de gradient stochastique moyennée couplée avec une évolution de α suivant la relation $\alpha_{i+1} = \alpha_0(1 + \gamma \cdot i)^{-1}$, tandis que Loshchilov et Hutter [106] utilisent une variante du recuit simulé. Toutefois, cela nécessite une intervention manuelle lors de la configuration préalable des hyperparamètres additionnels. Plusieurs travaux se sont donc penchés sur des méthodes avec moment dites adaptatives, dans lesquelles α s'ajuste automatiquement durant l'apprentissage en fonction de diverses heuristiques [42, 167, 186, 81].

Indépendamment de l'algorithme de descente de gradient utilisé, un point crucial dans l'optimisation des réseaux profonds réside dans le choix des poids initiaux. L'*initialisation* de ceux-ci conditionne également la capacité de la descente de gradient à converger vers un optimal de bonne qualité. Si, comme nous avons pu le voir, les méthodes de préentraînement non-supervisées ont été plébiscitées par le passé [66, 10], les réseaux sont dorénavant directement entraînés de façon supervisée. L'idée fondamentale des stratégies d'initialisation des poids consiste à leur affecter des valeurs aléatoires limitant l'explosion et l'évanescence des gradients. Glorot et Bengio [55] et He et al. [63] proposent ainsi une initialisation permettant d'obtenir des activations initiales normalement distribuées, ce qui facilite l'apprentissage en

garantissant un flot raisonnable des gradients. Dans la même veine, Saxe, McClelland et Ganguli [148] initialisent les noyaux de convolution de leurs réseaux à l'aide de matrices orthogonales aléatoires afin, d'une part, de conserver constante la norme des activations d'une couche à l'autre et, d'autre part, de décorréler entre eux les filtres initiaux.

Compte tenu de la nature stochastique de l'optimisation des réseaux de neurones, la communauté a consolidé un certain nombre de bonnes pratiques empiriques [93, 9, 16]. Comme pour l'apprentissage automatique des modèles non profonds, il est recommandé de normaliser les données d'entrée. Généralement, les images sont ainsi normalisées en soustrayant la valeur du pixel moyen calculé sur l'ensemble du jeu de données. Dans certains cas, notamment pour des images présentant une structure très similaire, c'est l'image moyenne qui est soustraite. Il est plus rare d'appliquer une normalisation sur la variance.

Lors de l'apprentissage, il est recommandé de mélanger les données après chaque passe sur le jeu d'apprentissage, afin d'éviter des cycles dans la descente de gradient [93]. En outre, la taille des mini-lots influe sur la descente de gradient. Plus les mini-lots sont grands, plus la descente est stable, mais une taille de mini-lots faible introduit une stochasticité plus forte dans la descente de gradient qui peut être bénéfique pour la généralisation du modèle. Enfin, il est recommandé d'accélérer la descente de gradient en démarrant avec un taux d'apprentissage élevé et de le réduire par la suite [9]. Les hyperparamètres de la descente de gradient sont souvent délicats à régler de façon optimale, mais il est possible de les valider sur un sous-ensemble du jeu de données restreint pour les généraliser sur l'ensemble de la base de données [16]. L'arrêt de l'apprentissage se fait généralement quand l'erreur de validation a cessé de décroître, ou à défaut lorsque l'erreur d'apprentissage ne diminue plus, au risque d'un surapprentissage.

Les bibliothèques logicielles récentes d'apprentissage profond implémentent pour la plupart ce type de bonnes pratiques, ainsi que les régularisations, initialisations, politiques d'évolution du taux d'apprentissage et variantes de la descente de gradient. Ceci simplifie grandement le travail d'expérimentation et diminue la part d'incertitude due à des pratiques divergentes au sein de la communauté. Pourtant, le réglage des hyperparamètres conserve une influence considérable dans les performances des différents modèles. L'absence d'études statistiques de robustesse, comme la répétition des entraînements et le moyennage des résultats, conduit parfois à conclure erronément sur les performances comparatives des modèles [125], bruitées par l'influence hasardeuse des hyperparamètres d'optimisation.

Il est important de noter que la descente de gradient minimise l'erreur sur le jeu d'entraînement, mais l'erreur réellement intéressante est celle commise sur les données réelles. Autrement dit, l'apprentissage minimise un risque empirique qui ne correspond pas nécessairement au risque réel. Or cette erreur est inaccessible, l'ensemble des données réelles étant infini et non-annoté. Nous devons donc nous contenter du risque empirique mesurable, au prix d'un surapprentissage occasionnel. En effet, dans certains cas le modèle peut apprendre des connaissances biaisées liées au choix des exemples du jeu d'entraînement ne se généralisant pas aux données réelles. Par exemple, un modèle devant discriminer entre des images de chats et des images de chiens entraîné sur des photos de chats prises majoritairement de jour et des photos de chiens prises majoritairement de nuit risquerait d'apprendre des caractéristiques liées à l'illumination plutôt qu'à l'espèce de l'animal. Il est important de noter que la descente de gradient minimise l'erreur sur le jeu d'entraînement, mais l'erreur réellement intéressante est celle commise sur les données réelles. Autrement dit, l'apprentissage minimise un risque empirique qui ne correspond pas nécessairement au risque réel. Or cette erreur est inaccessible, l'ensemble des données réelles étant infini et non-annoté. Nous devons donc nous contenter du risque empirique mesurable, au prix d'un surapprentissage occasionnel. En effet, dans certains cas le modèle peut apprendre des connaissances biaisées liées au choix des exemples du jeu d'entraînement ne se généralisant pas aux données réelles. Par exemple, un modèle devant discriminer entre des images de chats et des images de chiens entraîné sur des photos de chats prises majoritairement de jour

et des photos de chiens prises majoritairement de nuit risquerait d'apprendre des caractéristiques liées à l'illumination plutôt qu'à l'espèce de l'animal. Pour limiter ce phénomène de surapprentissage, il est possible de faire intervenir des techniques de *régularisation*. Celles-ci visent à contrebalancer la nature empirique de la fonction objectif et les biais inhérents au jeu de données. Une première méthode classique consiste à limiter l'amplitude des poids des connexions du réseau. Cette méthode, appelée *weight decay*, ou dégradation des pondérations, ajoute une pénalité au terme d'erreur globale dépendante de la norme des poids. Ainsi, la fonction de coût totale devient :

$$\mathcal{L}_{totale} = \mathcal{L}_{coût}(\mathrm{W}, \Omega) + \lambda \sum_{w \in \mathrm{W}} w^2 \ .$$

Krogh et Hertz [85] ont montré que cette simple pénalité permet de réduire l'erreur de généralisation du modèle.

Plus récemment, le *Dropout* [160] a été proposé comme méthode de régularisation pour lutter contre le surapprentissage. Les réseaux de neurones contenant de nombreux paramètres, l'idée est d'aléatoirement éteindre certains neurones à chaque itération de la phase d'apprentissage avec une probabilité p. Toutes les connexions liées aux neurones éteints sont alors neutralisées et seuls les poids du réseau réduit sont mis à jour lors de la descente de gradient pour cette itération. Lors de la phase d'inférence, toutes les activations liées à ces neurones sont pondérées par p afin de conserver la somme du signal constante. Chaque nœud du réseau ne voit ainsi qu'une partie du jeu de données, ce qui contraint les représentations internes à être redondantes pour conserver leur pouvoir discriminant. Les signaux faibles liés au biais intrinsèque du jeu de données ne peuvent alors être modélisés, palliant ainsi les problèmes de surapprentissage. Envisagé sous un angle différent, le *Dropout* peut être considéré comme une méthode générant un grand nombre de sous-réseaux entraînés en parallèle. En effet, si à chaque itération les neurones sont supprimés avec une probabilité $p = 0,5$, cela est équivalent à entraîner aléatoirement un réseau parmi les 2^n réseaux réduits possibles, n étant le nombre de paramètres sujets au *Dropout*. Lors de l'inférence, un seul réseau est utilisé, correspondant à la moyenne de ces réseaux réduits. Il s'agit ainsi d'une forme de régularisation par apprentissage par ensemble. D'autres régularisations s'inspirent de cette technique, comme le *DropConnect* [178], supprimant des synapses plutôt que des neurones, ou l'échantillonnage stochastique de Zeiler et Fergus [187].

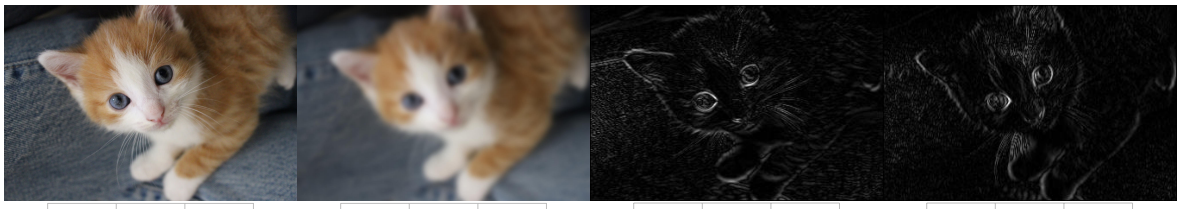
Une méthode alternative pour lutter contre le surapprentissage consiste à alimenter le jeu de données d'entraînement en exemples synthétiques. En augmentant artificiellement le nombre d'échantillons d'apprentissage, il est possible d'augmenter la variété des exemples auxquels le modèle est exposé et donc de réduire le biais intrinsèque du jeu de données. On parle alors d'*augmentation de données*. Dans le cas des images, cela passe le plus souvent par des transformations géométriques qui n'affectent pas leur sémantique, comme des symétries gauche-droite, des rotations ou encore des redimensionnements.

Enfin, la normalisation par lot, ou *Batch Normalization* (BN), [77] est parfois présentée comme une méthode de régularisation, en ce que les moments statistiques qui y sont estimés le sont de façon stochastique, ce qui ajoute un faible bruit aux représentations internes à chaque couche.

2.1.4 Réseaux de neurones convolutifs profonds

Si le principe de partage des poids pour effectuer de façon dense la même opération sur l'ensemble de l'image remonte au *Neocognitron* [52], la notion de couche de convolution est due à LeCun et al. [94]. Le produit de convolution entre deux fonctions f et g est un opérateur commutatif et bilinéaire, habituellement noté $f * g$, qui s'obtient par la formule suivante :

$$(f * g)(x) = \int_{-\infty}^{+\infty} f(t)g(x-t)\mathrm{d}t \ . \tag{2.5}$$



0	0	0
0	1	0
0	0	0

Image originale (identité)

0,25	0,5	0,25
0,5	1	0,5
0,25	0,5	0,25

Filtre gaussien (flou)

+1	+2	+1
0	0	0
-1	-2	-1

Filtre de Sobel [158] (contours horizontaux)

+1	0	-1
+2	0	-2
+1	0	-1

Filtre de Sobel [158] (contours verticaux)

Figure 2.5 – Exemples de filtrages par différents noyaux de convolution.

Les convolutions sont extrêmement populaires en traitement du signal compte-tenu de l'omniprésence de la transformée de Fourier en analyse harmonique [47]. En effet, la transformation de Fourier $\mathcal{F}$, ransporte les convolutions dans l'espace réel en multiplications dans l'espace spectral, et inversement :

$$\mathcal{F}(f * g) = \mathcal{F}(f)\mathcal{F}(g)\,. \tag{2.6}$$

En traitement d'images, la convolution discrète intervient dans le calcul des gradients utilisés par les descripteurs SIFT [107] et HOG [35]. Les filtres convolutifs sont également à la base de la théorie des ondelettes [110] et des ses applications en compression d'image avec le format JPEG [37] et en détection de visages Viola et Jones [176] par le biais des caractéristiques de pseudo-Haar [131]. Les neurosciences ont par ailleurs mis en évidence des similarités notables entre le modèle de filtre de Gabor, couramment employé pour le calcul de caractéristiques d'image[133], et les réponses neuronales du cortex visuel chez les mammifères [111, 80]. La Figure 2.5 illustre le résultat de filtrages classiques, comme le calcul des gradients discrets par filtre de Sobel [158] et l'application d'un flou grâce à un noyau gaussien. Il existe en outre des opérations de filtrage non-linéaire, comme le débruitage par filtre médian [49], qui ne peuvent pas s'écrire sous forme de convolution.

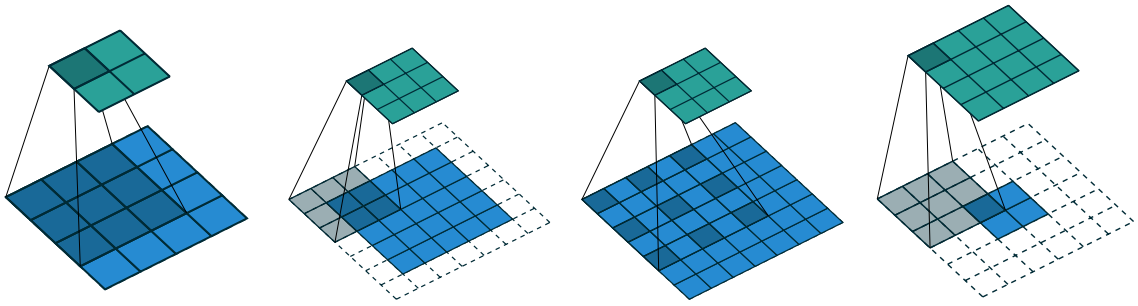
L'idée de LeCun et al. [94] est de remplacer les premières couches d'un réseau de neurones par des couches convolutives. Les neurones sont regroupés localement et calculent chacun une convolution sur une partie de l'image. Pour simplifier le modèle, les poids sont partagés, c'est-à-dire que tous les groupes de neurones calculent la même convolution. Plusieurs convolutions peuvent néanmoins être calculées en parallèle. Les noyaux de convolution étant optimisés durant l'apprentissage, l'apprentissage par représentation va se faire naturellement dans le domaine image à l'aide d'opérateurs adaptés. Yosinski et al. [184] ont notamment constaté que la première couche convolutive d'un réseau profond tend à s'approcher systématiquement de filtres de Gabor [184].

Convolution

Dans le cas discret, le produit de convolution se réécrit :

$$(f * g)[n] = \sum_{k=-\infty}^{+\infty} f[n]g[k-n] \tag{2.7}$$

Cette formulation s'intéresse cependant aux signaux 1D, tandis que les images sont des signaux 2D. La convolution discrète s'étend sans problème aux fonctions multivariées.



Convolution | Convolution avec pas et padding | Convolution à trous | Déconvolution

FIGURE 2.6 – Opérateur de convolution et variantes sur une image (figures extraites de [43]).

En deux dimensions, si l'on note $\mathrm{I} : [\![1;w]\!] \times [\![1;h]\!] \to \mathbb{R}$ une image I de taille $w \times h$ et $\mathrm{K} : [\![1;k_w]\!] \times [\![1;k_h]\!] \to \mathbb{R}$ le *noyau* de convolution de dimension $k_w \times k_h$, alors on définit le filtre $\mathcal{K}$ tel que :

$$\mathcal{K}(\mathrm{I})[m,n] = \mathrm{K} * \mathrm{I}[m,n] = \sum_{i=-p}^{+p} \sum_{j=-q}^{+q} \mathrm{I}[m-i, n-j] \cdot \mathrm{K}[i,j] \,, \tag{2.8}$$

avec $p = \frac{k_w - 1}{2}$ et $q = \frac{k_h - 1}{2}$. Ce calcul est illustré dans la Figure 2.6.

Un des inconvénients de ce produit est que le noyau de convolution K et l'image I sont parcourus en sens inverse, les indices de l'un augmentant tandis que les indices de l'autre décroissent. En pratique, la plupart des bibliothèques implémentent l'opérateur de *corrélation croisée* :

$$\mathcal{K}(\mathrm{I})[m,n] = \mathrm{K} \star \mathrm{I}[m,n] = \sum_{i=-p}^{+p} \sum_{j=-q}^{+q} \mathrm{I}[m+i, n+j] \cdot \mathrm{K}[i,j] \,. \tag{2.9}$$

Cet opérateur perd la commutativité mais est plus simple à programmer. Les paramètres de K étant optimisables, il est équivalent en pratique d'utiliser une corrélation croisée ou une convolution, car leurs matrices sont identiques à symétrie près. Les autres opérations intervenant dans des CNN n'étant pas commutatives, la perte de cette propriété n'a donc que peu d'importance. Dans la suite, les formules seront données pour l'opérateur de convolution.

La corrélation croisée et la convolution souffrent toutes deux d'une inconnue lorsque l'opérateur agit sur les bords, puisque les valeurs de I hors de l'image sont indéfinies. En règle générale, on ne calcule pas ces valeurs et les lignes et colonnes pour lesquelles le produit de convolution est indéfini sont ignorées, ce qui réduit la taille effective de l'image. On parle alors de corrélation croisée *valide*. Il est également possible de remplir les valeurs manquantes de I par des zéros (*zero-padding*) (cf. Figure 2.6), pour un nombre de lignes et de colonnes égal à la moitié de la taille du noyau de convolution dans chaque direction. On parle alors de corrélation croisée *identique*, car le résultat du filtrage est de même dimension que l'image d'entrée. Enfin, il est possible de remplir les valeurs manquantes par autant de zéros que nécessaire pour que chaque élément de I soit visité par chaque élément de K, auquel cas on parle de corrélation croisée *complète*.

En pratique, une couche de convolution en dimension n d'un réseau de neurones est paramétrée par :

- Les dimensions $(k_1, \ldots, k_n)$ des noyaux de convolution, généralement identique selon toutes les dimensions,
- Le nombre C de convolutions parallèles, qui définit le nombre de cartes d'activations en sortie de couche,
- Le pas s de la convolution,
- Le *padding* p.

Ainsi, une couche de convolution possède $k_1 \times \cdots \times k_n \times \mathrm{C}$ paramètres optimisables. Dans le cas le plus courant de la dimension 2, les noyaux de convolution sont généralement carrés, c'est-à-dire qu'une couche de convolution 2D contient $\mathrm{C}k^2$ paramètres.

L'intérêt de la convolution dans les réseaux de neurones profonds est triple [58] :

— Les interactions convolutives sont parcimonieuses, la taille des noyaux de convolution étant très faible devant la taille des images,
— Les caractéristiques extraites par convolution sont équivariantes aux translations de l'image, c'est-à-dire qu'une translation de l'image d'entrée translate les cartes d'activation de la même façon,
— Les paramètres de la convolution sont partagés pour l'ensemble de l'image, ce qui permet de détecter les mêmes caractéristiques peu importe leur position dans l'image avec un très faible coût de stockage en mémoire des paramètres.

Comparée à une couche entièrement connectée, la couche convolutive n'est pas invariante à la permutation des pixels car elle possède un *a priori* fort sur la structure spatiale des données. Cet *a priori* est lié à la notion d'équivariance sémantique des images par rapport à certaines transformations géométriques. Néanmoins, il faut garder à l'esprit que cette connaissance structurelle n'est pas toujours respectée. Dans une série temporelle, l'apparition d'une anomalie peut avoir un sens différent en fonction du moment auquel elle se produit. À l'inverse, la convolution 1D part du principe que l'anomalie excitera les neurones de la même façon quelle que soit sa position dans le temps. Cet *a priori* fort est bien adapté aux images, et tout particulièrement aux images aériennes et satellitaires, qui présentent des régularités spécifiques qui seront détaillées plus tard. La structure même des CNN est donc adaptée au traitement d'images, qu'ils permettent de décomposer dans un espace de représentation doté d'une équivariance forte à diverses transformations [171].

Conventionnellement, en 2D, on représente les cartes d'activation des neurones, ou cartes de caractéristiques, sous la forme de tenseurs de dimension 3 $(\mathrm{C}, \mathrm{W}, \mathrm{H})$ avec C le nombre de canaux, également appelé nombre de plans de convolutions, W la largeur et H la hauteur des cartes.

Une couche convolutive combine les n_{in} cartes d'activation d'entrée avec le j^{e} noyau de convolution K_j :

$$\forall j \in [\![1; n_{out}]\!], \quad o_j = b_j + \sum_{i=1}^{n_{in}} \mathrm{K}(z_i)\,, \tag{2.10}$$

c'est-à-dire :

$$\forall j \in [\![1; n_{out}]\!], \quad o_j(m,n) = b_j + \sum_{i=1}^{n_{in}} \sum_{p=-\frac{k-1}{2}}^{+\frac{k-1}{2}} \sum_{q=-\frac{k-1}{2}}^{+\frac{k-1}{2}} z_i(m-p, n-q) \cdot k_j(p,q)\,. \tag{2.11}$$

Ainsi, une convolution transforme un tenseur $(\mathrm{C}_{in}, \mathrm{W}_{in}, \mathrm{H}_{in})$ en tenseur $(\mathrm{C}_{out}, \mathrm{W}_{out}, \mathrm{H}_{out})$ dont les dimensions spatiales se calculent par [4] :

$$out = in - kernel + 2 \cdot padding + 1\,. \tag{2.12}$$

Convolution à pas Une première variante du produit de convolution consiste à sous-échantillonner virtuellement la dimension des cartes d'activation produites d'un facteur s. Pour ce faire, il suffit de ne visiter les éléments de I qu'avec un pas de s :

$$\mathcal{K}_s(\mathrm{I})(m,n) = \mathrm{K}_s \star \mathrm{I} = \sum_{i=-p}^{+p} \sum_{i=-q}^{+q} \mathrm{I}[s \cdot m + i, s \cdot n + j] \cdot \mathrm{K}[i,j]\,. \tag{2.13}$$

4. Les équations d'arithmétique des convolutions sont tirées de Dumoulin et Visin [43].

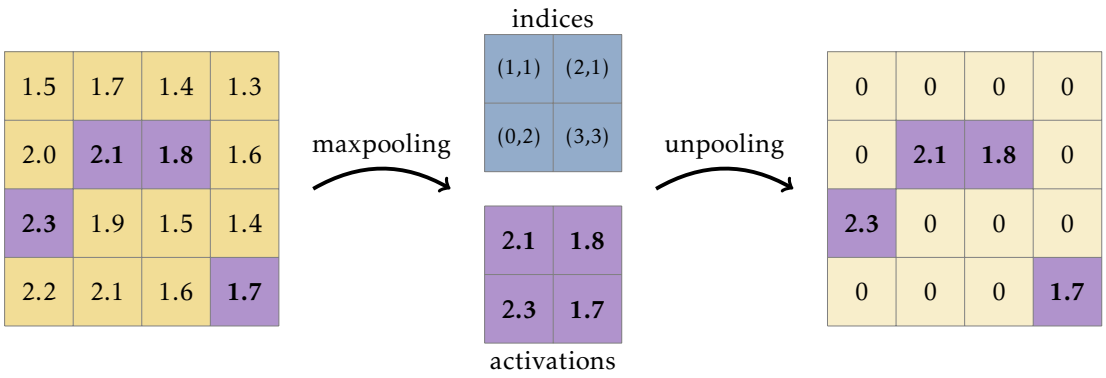


FIGURE 2.7 – Sous-échantillonnage et sur-échantillonnage par valeurs maximales en deux dimensions.

Une convolution à pas transforme donc un tenseur (C_{in}, W_{in}, H_{in}) en tenseur $(C_{out}, W_{out}, H_{out})$ avec les dimensions spatiales se calculant par :

$$out = \left\lfloor \frac{in - kernel + 2 \cdot padding}{s} \right\rfloor + 1 \,. \tag{2.14}$$

Convolution à trous La convolution à trous, ou convolution dilatée [185][5], consiste à réaliser une convolution en observant I à une résolution plus faible que sa résolution réelle, en sautant certaines de ses valeurs. Le noyau de convolution est ainsi virtuellement dilaté d'un facteur d, les valeurs manquantes étant remplacées par des 0. En pratique, la convolution à trous se calcule par la formule :

$$\mathcal{K}^{(d)}(\mathrm{I})(m,n) = \mathrm{K}_s \star \mathrm{I} = \sum_{i=-p}^{+p} \sum_{i=-q}^{+q} \mathrm{I}[m + d \cdot i, n + \cdot j] \cdot \mathrm{K}[i,j] \,. \tag{2.15}$$

Les cartes d'activation en sortie d'une convolution à trous ont pour dimensions :

$$out = \left\lfloor \frac{in - kernel - (kernel - 1)(dilation - 1) + 2 \cdot padding}{s} \right\rfloor + 1 \,. \tag{2.16}$$

Convolution transposée La convolution transposée est une opération s'opposant à la convolution traditionnelle en ce qu'elle correspond à son gradient par rapport à ses entrées. Pour un noyau de convolution k donné, la convolution transposée permet de reconstruire une image I à partir des activations Z, dont les dimensions s'obtiennent par

$$out = (in - 1) \cdot s + kernel - 2 \cdot padding \,. \tag{2.17}$$

Plus prosaïquement, il est possible d'envisager la convolution transposée comme une convolution à pas fractionnel, c'est-à-dire une convolution de pas $s = \frac{1}{s'}$ avec $s' \in \mathbb{N}^*$.

Cette convolution est parfois appelée à tort « déconvolution » dans la littérature, sans toutefois correspondre à l'opérateur mathématique éponyme, défini comme l'inverse de l'opérateur de convolution. La convolution transposée est particulièrement utile pour inverser les effets d'une couche convolutive, par exemple dans la phase de décodage d'un auto-encodeur convolutif [190] ou pour la super-résolution [41].

Échantillonnage

Sous-échantillonnage Afin de réduire la dimension des cartes d'activation dans le réseau, il est utile d'opérer des sous-échantillonnages. Il s'agit généralement d'appliquer un filtre

5. L'algorithme à trous [153] applique un même filtre à plusieurs échelles en utilisant des convolutions dilatées. La différence entre les deux est subtile et ne sera pas discutée ici.

par fenêtre glissante non-recouvrante sur les données d'entrée. Ce filtre est en règle générale l'opérateur max ou l'opérateur de moyenne sur une fenêtre de taille fixe. On parle alors de *max pooling* ou d'*average pooling* [193]. Un exemple de sous-échantillonnage en 2D est illustré dans la Figure 2.7. Dans certains cas, la taille de la fenêtre de sous-échantillonnage n'est pas définie à l'avance mais la dimension des cartes d'activation de sortie l'est. On parle alors de sous-échantillonnage adaptatif. Il est utilisé dans certains réseaux pour réduire brutalement la dimension des cartes d'activation lorsque la taille des images est arbitraire. Dans le cas contraire, les dimensions des couches entièrement connectées déterminent la taille des images d'entrée. Outre la réduction de dimension, l'intérêt de ces opérateurs est d'introduire une invariance aux déformations locales.

Les dimensions d'une carte d'activation en sortie d'un sous-échantillonnage sont :

$$out = \left\lfloor \frac{in - kernel}{s} \right\rfloor + 1 \ .$$

Le sous-échantillonnage n'a pas de paramètre optimisable, il s'agit d'une fonction complètement déterminée.

Sur-échantillonnage Le sur-échantillonnage, ou *unpooling*, est l'opération inverse du sous-échantillonnage et tente de reconstruire une entrée à partir de sa sortie. Le sous-échantillonnage étant une opération perdant de l'information, le sur-échantillonnage est approximatif. Dans le cas du sur-échantillonnage par valeur moyenne, la même valeur sera répliquée plusieurs fois dans l'image à résolution augmentée. Dans le cas du sur-échantillonnage par valeur maximale, le maximum sera replacé à sa position initiale et les valeurs restantes complétées par des zéros, comme illustré dans la Figure 2.7. Les dimensions d'une carte d'activation en sortie d'un sur-échantillonnage sont :

$$out = (in - 1) \cdot s + kernel \ .$$

Comme pour le sous-échantillonnage dont il est la transposée, le sur-échantillonnage ne comprend aucun paramètre optimisable.

Normalisation

Si la normalisation des données afin de centrer et rendre unitaire les distributions statistiques est une pratique ancienne, l'utilisation de couches de normalisation des activations au sein-même des réseaux profonds est relativement récente. Le principe de la normalisation est d'appliquer une transformation aux cartes d'activation afin de leur imposer des propriétés statistiques particulières bénéfiques pour l'optimisation du modèle.

Jarrett et al. [78] propose ainsi une normalisation locale du contraste des cartes d'activation après les couches convolutives en s'inspirant de modèles biologiques [136]. Pour un tenseur de N cartes d'activations $a_1, \ldots, a_N$, les cartes normalisées z_i s'obtiennent d'abord par soustraction d'une valeur moyenne locale sur une fenêtre spatiale gaussienne :

$$b_i[x,y] = a_i[x,y] - \sum_{p,q} w_{p,q} \cdot a_i[x+p, j+q]$$

avec $w_{p,q}$ une fenêtre gaussienne telle que $\sum_{p,q} w_{p,q} = 1$ puis par normalisation des amplitudes par l'écart-type pondéré des caractéristiques sur une fenêtre spatiale :

$$z_i[x,y] = \frac{b_i[x,y]}{\max(\text{moy}(\sigma[x,y]), \sigma[x,y])}$$

avec $\sigma[x,y] = \left(\sum_{p,q} w_{p,q} \cdot b_i^2[x+p, y+q]\right)^{\frac{1}{2}}$.

L'article séminal de Krizhevsky, Sutskever et Hinton [84] introduit également une couche de normalisation locale, *Local Response Normalization* (LRN), qui inhibe le signal des cartes d'activations adjacentes à celle d'un neurone fortement excité. Ce procédé s'inspire de l'inhibition latérale existant dans les neurones biologiques et permet d'améliorer la généralisation du modèle. Contrairement à la normalisation du contraste de Jarrett et al. [78], cette normalisation ne s'applique pas sur un voisinage spatial, mais sur les cartes d'activations voisines de celle considérée, dans la troisième direction du tenseur. Cette normalisation s'écrit sous la forme d'un noyau appliqué aux cartes d'activation 2D. Pour un tenseur de N cartes d'activations $a^1, \ldots, a^N$, les cartes normalisées z^i s'obtiennent par :

$$z^i[x,y] = \frac{a^i[x,y]}{\left(k + \alpha \sum_{j=max(0,i-n/2)}^{min(N-1,i+n/2)} (a^j[x,y])^2\right)^\beta} \quad .$$

avec n la taille du voisinage à considérer (dans la direction des canaux). Cette opération vient ainsi normaliser l'intensité du vecteur d'activations $(a^1[x,y], a^2[x,y], \ldots, a^N[x,y])$ à chaque position spatiale de la carte de caractéristiques.

La normalisation des activations la plus courante dans l'état de l'art est la BN [77]. Ce procédé consiste à normaliser les moments statistiques des cartes d'activation plan par plan. Il suppose un entraînement appliquant l'algorithme de descente de gradient stochastique par mini-lots. Pour un mini-lot de taille N, le réseau opère à chaque itération sur un tenseur (N, C, W, H) de la façon suivante :

Définition 5. *Algorithme de normalisation par mini-lot :*

Notons $a_i^{(n)}, i \in [\![1, C]\!]$ les activations du n^e plan en sortie d'une couche donnée. Durant la phase d'apprentissage, la moyenne μ et la variance σ^2 au sein d'un mini-lot sont calculées à la volée :

$$\mu_i = \frac{1}{N} \sum_{n=1}^{N} a_i^{(n)} \quad et \quad \sigma_i^2 = \frac{1}{N} \sum_{n=1}^{N} (a_i^{(n)} - \mu_i)^2$$

Les valeurs moyennées de μ et σ^2 sur l'ensemble du jeu de données sont conservées en mémoire et ré-utilisées en phase d'inférence afin de rendre les calculs indépendants de la taille du mini-lot.

Dans tous les cas, les activations normalisées $\hat{a}$ sont obtenues par :

$$\hat{a}_i = \frac{a_i - \mu_i}{\sqrt{\sigma_i^2 + \epsilon}}$$

.

La normalisation par lot est généralement suivie d'une transformée affine $z_i = \alpha \hat{a}_i + \beta$. [6] La version proposée ici s'applique au cas 2D, mais s'étend également aux cas des activations 1D et 3D.

Cette normalisation permet de rendre le flot des gradients pendant la rétro-propagation indépendante de la variance des poids de chaque couche, ce qui permet l'optimisation de réseaux arbitrairement profonds. En pratique, l'utilisation de la BN permet d'améliorer les performances de classification des modèles et accélère significativement leur vitesse de convergence en lissant la surface de la fonction de coût [147]. Cette normalisation se retrouve dans la grande majorité des architectures de réseaux profonds actuelles.

La fonction de transfert non-linéaire des neurones s'applique en sortie de convolution avant ou après la normalisation selon les modèles.

6. Dans ce cas, la BN contient 2C paramètres optimisables.

Couche entièrement connectée

Une couche entièrement connectée correspond à un graphe biparti complet, au sein duquel tous les neurones d'entrée sont connectés par des synapses à tous les neurones de sortie. Cela correspond de fait au perceptron de Rosenblatt [142]. La non-linéarité est appliquée sous la forme d'une fonction d'activation sur le vecteur de sortie. Une couche entièrement connectée peut se conceptualiser comme une simple multiplication matricielle, transformant un vecteur de dimensions $1 \times N$ en vecteur de dimensions $1 \times M$ par le biais d'une matrice de poids $N \times M$. Lorsqu'il est utilisé, le *Dropout* [160] est généralement appliqué sur les poids des couches entièrement connectées. En effet, ces couches contiennent un nombre important de paramètres et sont sensibles au surapprentissage. À l'inverse, les couches convolutives sont rarement soumises au *Dropout*, car la suppression stochastique de neurones risque d'avoir des effets négatifs sur la structure spatiale des cartes d'activation.

2.2 Apprentissage profond pour la segmentation sémantique

2.2.1 De la classification à la segmentation

La segmentation d'image est une des premières tâches envisagées dans le cadre de la vision artificielle. La légende veut que Minsky en 1964 ait demandé à son étudiant Gerald Sussman de « passer l'été à connecter une caméra à un ordinateur et réussir à faire en sorte que l'ordinateur décrive ce qu'il voit » [7]. La tâche envisagée par Minsky et Papert pour le *Summer Vision Project* (cf. Figure 2.1) consistait notamment à « construire un système de programmes divisant une image issue d'un tube dissecteur [8] en différentes régions telles que : plutôt des objets, plutôt l'arrière-plan, ou du chaos » avec pour objectif final un logiciel réalisant « l'identification d'objets, qui nommera chaque objet en les faisant correspondre avec un vocabulaire d'objets connus. » [132] Dès le départ, la reconnaissance de formes s'intéresse donc au découpage sémantique des images afin de comprendre les scènes visuelles qu'elles représentent (cf. Figure 2.8). Si cette tâche paraît triviale pour un humain, elle représente pourtant un défi considérable pour la machine. L'équipe de Minsky se heurte rapidement au paradoxe de Moravec [118] : « il est relativement aisé de mettre des ordinateurs au niveau d'un humain adulte dans le cadre d'un test d'intelligence ou d'une partie de dames, mais difficile voire impossible de leur donner les capacités de perception et la mobilité d'un bébé » [9].

De nombreux travaux se sont dès lors penchés sur le problème de la reconnaissance d'objet, c'est-à-dire l'identification d'un objet présent dans une image. On parlera ici de classification d'images, une tâche consistant à associer une image à un type d'objet. Cette tâche a concentré la majorité des efforts de la communauté, de l'utilisation de descripteurs *ad hoc* [170] aux caractéristiques apprises [175] en passant par les modèles probabilistes [149]. L'arrivée récente de grandes bases de données d'images annotées comme CIFAR-10 et CIFAR-100 [83], puis ImageNet [40, 145] ont notamment permis d'aboutir au succès des réseaux convolutifs profonds en classification de chiffres sur la base de données MNIST [94], en reconnaissance de panneaux de signalisation [161], en identification de caractères chinois [100] et bien sûr en reconnaissance d'objets en tout genre sur ImageNet [84].

L'architecture LeNet-5, développée par LeCun et al. [94], définit la structure de référence d'un CNN. Elle consiste en deux couches convolutives suivies de trois couches entièrement connectées, comme illustré dans la Figure 2.9. La partie convolutive du modèle réalise

7. *"spend the summer linking a camera to a computer and getting the computer to describe what it saw"*, tel que rapporté par Szeliski [166], citant lui-même Boden [12] reprenant Crevier [32].

8. En référence au *vidisector*, le dissecteur d'images inventé par Philo Farnsworth en 1927 sur le principe du tube cathodique. L'appareil reçoit de la lumière, ce qui stimule une photocathode et émet des électrons, produisant un signal électrique permettant de représenter l'image. C'est l'inverse de l'écran de télévision.

9. *"it is comparatively easy to make computers exhibit adult level performance on intelligence tests or playing checkers, and difficult or impossible to give them the skills of a one-year-old when it comes to perception and mobility"*



FIGURE 2.8 – Exemple de classification et de segmentation sur une même image. La classification s'intéresse à la reconnaissance d'objet pour toute l'image, tandis que la segmentation opère sur chaque pixel. Crédits image : PIRO4D (CC0).

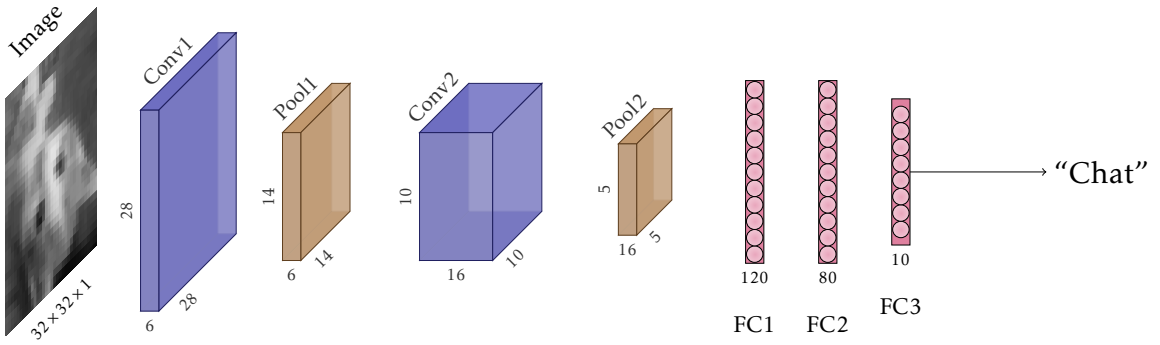


FIGURE 2.9 – Architecture LeNet-5 [94].

l'extraction de caractéristiques dans le domaine image, tandis que les couches entièrement connectées forment un perceptron multicouche opérant sur une représentation vectorielle et réalisant la classification finale. Initialement, LeNet-5 a été développé pour la reconnaissance de chiffres manuscrits dans des images en niveaux de gris de dimensions 32×32. En comparaison de la taille de l'image, les filtres convolutifs utilisés sont relativement grands, les noyaux de convolution étant de taille 5×5. La première couche C1 comporte 6 noyaux, c'est-à-dire que six cartes d'activation sont générées. Elles sont sous-échantillonnées avec un pas de 2, puis transmises à la couche convolutive suivante C2. Chacun des 16 noyaux de convolution de la couche C2 produit une carte correspondant à la somme des plans de C1 filtrés par ce noyau. Les cartes sont à nouveau sous-échantillonnées, ce qui produit finalement 16 cartes d'activation 5×5. Celles-ci sont alors aplaties en un vecteur de taille 1×400, entièrement connecté à une première couche cachée de dimension 120, puis à une seconde de dimension 84. Finalement, ce descripteur est entièrement connecté au vecteur de sortie de taille 10, chaque activation correspondant à un des 10 chiffres possibles. La sortie est alors transformée par un *softmax* permettant de minimiser une entropie croisée par rétro-propagation du gradient.

La présence des couches entièrement connectées, dont le nombre de neurones est fixé, impose aux cartes d'activation issues des couches convolutives d'être de la bonne dimension. Cette contrainte impose en cascade que la taille des images traitées par LeNet soit toujours la même. Utiliser des images d'autres dimensions nécessiterait de ré-entraîner des couches

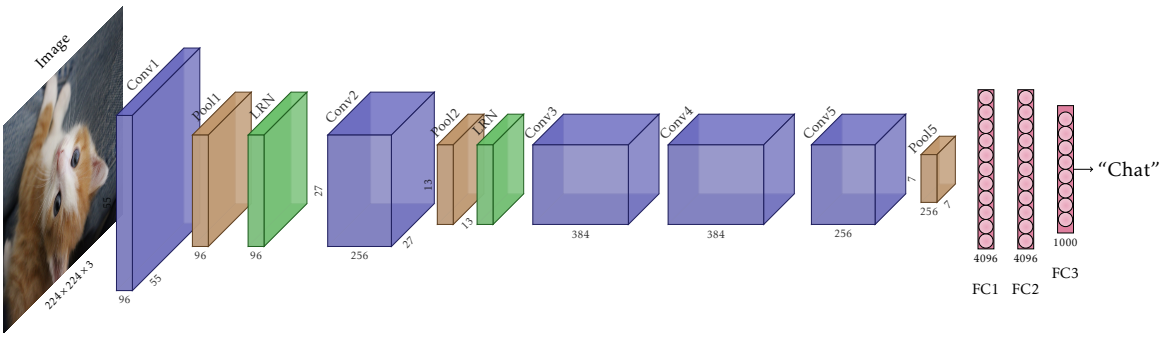


FIGURE 2.10 – Architecture AlexNet [84].

entièrement connectées avec le nombre adéquat de neurones. Ce défaut, dû à la présence des couches entièrement connectées, est partagé par la plupart des CNN.

Pour le traitement des images en couleur, le réseau AlexNet [84] utilise une approche similaire. Ce modèle, détaillé dans la Figure 2.10 comporte 8 couches, 5 convolutives et 3 entièrement connectées ; il traite des images rouge-vert-bleu (RVB) de dimensions 224×224. La première convolution utilise un grand noyau de dimensions 11×11 et précède un sous-échantillonnage, ce qui réduit fortement les dimensions de l'image. Ainsi, les cartes d'activation en sortie de la première couche sont de dimensions $96 \times 55 \times 55$, puis $256 \times 27 \times 27$ après la seconde. Une normalisation LRN est appliquée après les deux premières convolutions pour favoriser la généralisation du modèle. La structure convolutive d'AlexNet est conçue de telle sorte à ce que le nombre de cartes d'activation augmente de façon inversement proportionnelle à la réduction des dimensions spatiales. Cela permet de conserver un nombre raisonnable de valeurs à calculer tout en accroissant l'expressivité de la représentation apprise. Les trois couches de convolution suivantes produisent des tenseurs $384 \times 13 \times 13$ puis $256 \times 13 \times 13$, finalement sous-échantillonnés en 256 cartes de caractéristiques de dimensions 7×7, soit un vecteur unidimensionnel de longueur 12 544. Les couches entièrement connectées le réduisent alors à 2048 puis le projettent dans un vecteur de classification de taille 1000, correspondant aux classes d'ImageNet [40]. Ce modèle a permis à KRIZHEVSKY, SUTSKEVER et HINTON [84] de remporter la compétition ILSVRC [145] en 2012 avec un taux d'erreur de 15,3% en considérant les 5 prédictions les plus probables.

ZEILER et FERGUS [188] se sont penchés sur l'architecture AlexNet afin d'établir un diagnostic de ses points forts et de ses points faibles. En particulier, ils proposent d'inverser les opérations de convolution afin de pouvoir relier les représentations internes aux pixels de l'image originale. Ils définissent ainsi les opérations transposées de la couche de convolution et du sous-échantillonnage et construisent un *Deconvnet* permettant de visualiser les caractéristiques apprises par AlexNet. En outre, ils étudient attentivement les filtres convolutifs appris par les couches basses du réseau. Leurs travaux mettent en évidence plusieurs propriétés intéressantes. La première est que les caractéristiques des couches supérieures présentent une plus grande invariance aux transformations géométriques et colorimétriques de bas niveau, traduisant ainsi un niveau d'abstraction plus élevé. La seconde est que les deux premières couches convolutives d'AlexNet contiennent des filtres liés aux hautes et basses fréquences, mais conservent peu d'information des fréquences intermédiaires. Ils proposent ainsi de remplacer la première couche, dont les noyaux sont de dimension 11×11, par des filtres 7×7 appliqués sur l'image avec un pas de 2 plutôt qu'un pas de 4, afin de conserver plus d'information. Leurs filtres appris de cette façon présentent une meilleure variété et moins de filtres "morts" à très faible amplitude. Enfin, la visualisation des représentations internes permet de constater que celles-ci ne sont pas aléatoires, mais possèdent bien une sémantique liée à la discrimination des objets, certains neurones se focalisant par exemple sur les visages, d'autres sur les roues, etc.

Le modèle VGG-16 affine l'architecture en proposant notamment de réduire la dimen-

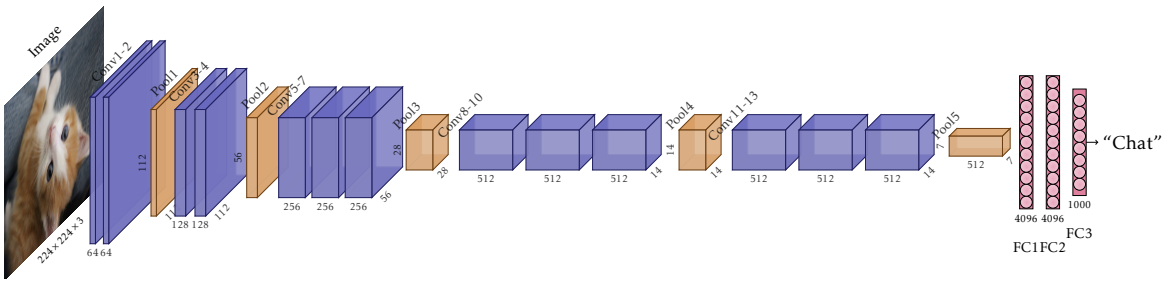


FIGURE 2.11 – Architecture VGG-16 [157].

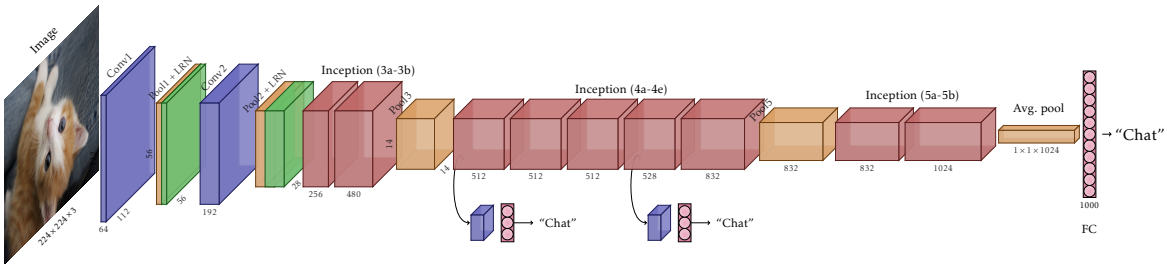


FIGURE 2.12 – Architecture GoogLeNet [163].

sion des noyaux de convolution. En effet, CHATFIELD et al. [21] et SIMONYAN et ZISSERMAN [157] suggèrent qu'il est plus simple d'optimiser plusieurs convolutions successives de noyaux 3×3 qu'une unique convolution de dimension 11×11. En outre, la présence de non-linéarités supplémentaires est susceptible d'accroître l'expressivité du modèle. Le modèle VGG-16 remplace donc chaque convolution large classique par un bloc de 2 ou 3 convolutions 3×3 successives, comme illustré par la Figure 2.11. Le modèle de référence VGG-16 comporte 16 couches dont 13 convolutives et 3 entièrement connectées, suivant l'approche canonique de LeNet et d'AlexNet. Il comporte 5 blocs convolutifs 3×3 chacun suivi d'un sous-échantillonnage de pas 2. Les deux premiers blocs comportent 2 couches de convolution, les trois suivants en comportant 3. Les cartes d'activation finales sont de dimension $512 \times 7 \times 7$, VGG-16 réalisant ainsi une réduction de dimension d'un facteur 32. Ce vecteur de longueur 25088 est ensuite réduit à 4096, puis aux 1000 classes d'intérêt d'ImageNet. Afin de limiter le surapprentissage, les couches entièrement connectées sont soumises au *Dropout*. Ces améliorations proposées à l'architecture classique des CNN ont permis d'obtenir en 2014 un taux d'erreur de 7,4% en reconnaissance d'objet durant la compétition ILSVRC.

Indépendamment, SZEGEDY et al. [163] proposent le modèle GoogLeNet avec 22 couches. Cette architecture introduit notamment le module *Inception* empilant plusieurs couches en parallèle, et plus seulement en profondeur. L'idée est de réaliser, pour une carte d'activation, l'extraction de caractéristiques à plusieurs niveaux de contexte en utilisant soit une convolution 1×1, c'est-à-dire une combinaison linéaire suivie d'une non-linéarité, soit un *pooling*, soit des convolutions 3×3 ou 5×5. Cela permet de coupler des caractéristiques dotées de l'invariance aux translations locales (issues du sous-échantillonnage) et des caractéristiques qui ne le sont pas, ce qui permet de gérer une plus grande variété de cas. Le module *Inception* est illustré dans la Figure 2.13 tandis que la Figure 2.12 détaille l'architecture complète du modèle GoogLeNet. Compte-tenu de la profondeur du réseau (22 couches), ses auteurs proposent de faciliter l'optimisation des couches les plus basses en ajoutant un classifieur au niveau des représentations intermédiaires, après les modules *Inception* (4a) et (4d). Cette approche profondément supervisée avait notamment déjà montré son efficacité pour lutter contre les problèmes de gradients évanescents [95]. Ce modèle obtient un taux d'erreur de seulement 6,4% en reconnaissance d'objets à l'ILSVRC. L'architecture GoogleNet est améliorée [164] par la suite en remplaçant les convolutions 5×5 du module *Inception* par

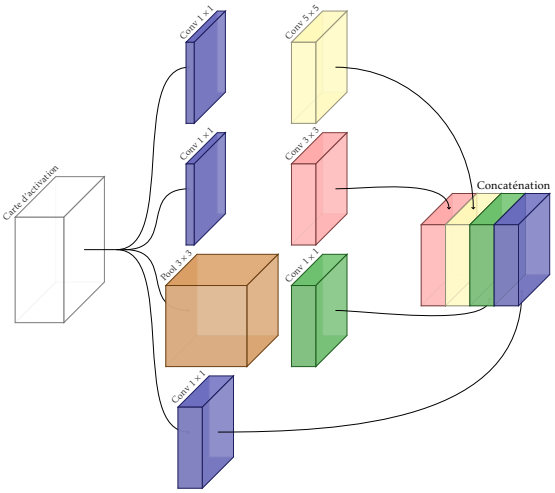


FIGURE 2.13 – Module *Inception* [163].

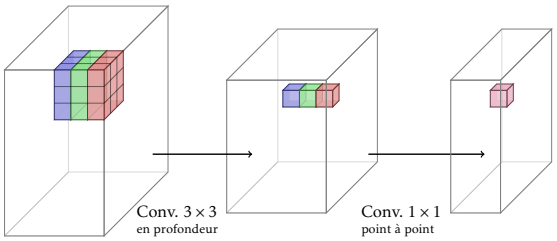


FIGURE 2.14 – Convolutions séparables en profondeur [25].

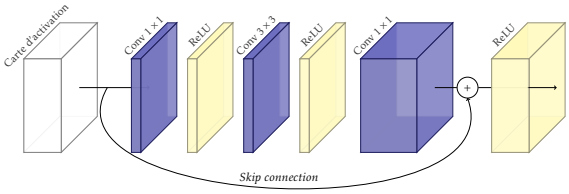


FIGURE 2.15 – Bloc convolutif résiduel [64].

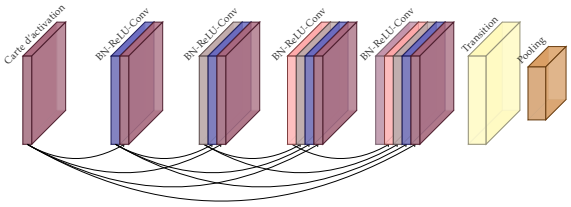


FIGURE 2.16 – Bloc convolutif dense [74].

deux convolutions 3×3, comme proposé dans le modèle VGG [157] et en intégrant la *Batch Normalization* [77].

En 2015, HE et al. [64] parviennent à obtenir un taux d'erreur de seulement 3,5% en reconnaissance d'objet durant ILSVRC. Leur approche consiste en un réseau très profond comprenant plus de 100 couches convolutives. L'optimisation est rendue possible d'une part grâce à la *Batch Normalization*, mais surtout grâce à l'apprentissage par résidu. L'idée est de briser la structure purement séquentielle des réseaux à propagation avant en ajoutant des connexions permettant de court-circuiter la couche suivante. Ces connexions, dites résiduelles, correspondent à une simple opération identité et permettent alors aux activations et au gradient de parcourir l'ensemble du réseau sans subir d'évanescence ou d'explosion dues à la règle de la dérivation en chaîne. Plutôt que de chercher à approcher $f : x \to f(x)$, le bloc résiduel va approcher $\hat{f} : x \to f(x) - x$, plus simple car d'amplitude a priori plus faible. Le bloc de convolution résiduel est illustré dans la Figure 2.15 et un exemple de modèle dit *ResNet* à 34 couches est détaillé dans la Figure 2.17. L'introduction de l'apprentissage par résidu change en partie le paradigme utilisé jusqu'alors pour la conception des CNN. Le bloc de base constitutif du réseau passe ainsi au bloc résiduel. Les ResNet possèdent beaucoup de couches mais comparativement peu de paramètres, car seule la dernière couche est entièrement connectée. Comme nous l'avons vu, les couches entièrement connectées concentrent en général la majorité des poids des CNN et sont également les plus sensibles au surapprentissage, nécessitant l'intégration de régularisations comme le *Dropout* [160]. ResNet ne contient quasiment que des convolutions 3×3, à l'exception de la première convolution 7×7 qui permet de réduire brutalement les dimensions spatiales de l'image. Il est intéressant de constater que la réduction de dimension finale avant la couche entièrement connectée se fait à l'aide d'un sous-échantillonnage adaptatif. Ainsi, peu importe la taille de l'image de l'entrée : le sous-échantillonnage moyennera les activations pour produire le vecteur de caractéristiques de taille attendue par la couche entièrement connectée. Cependant, il faut noter que le nombre élevé d'activations et de gradients intermédiaires à calculer rend les

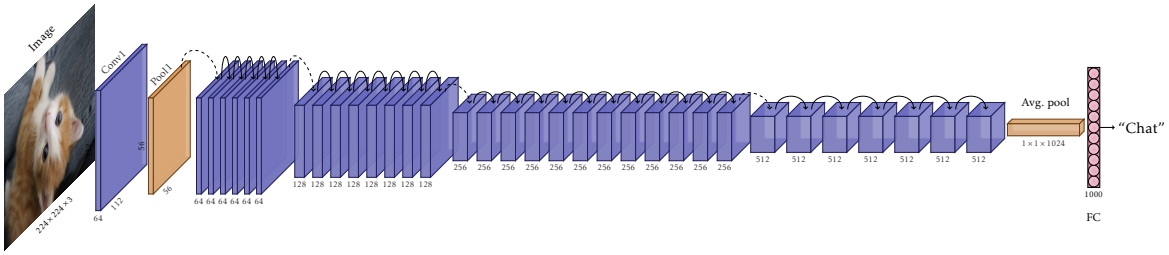


FIGURE 2.17 – Architecture ResNet-34 [64].

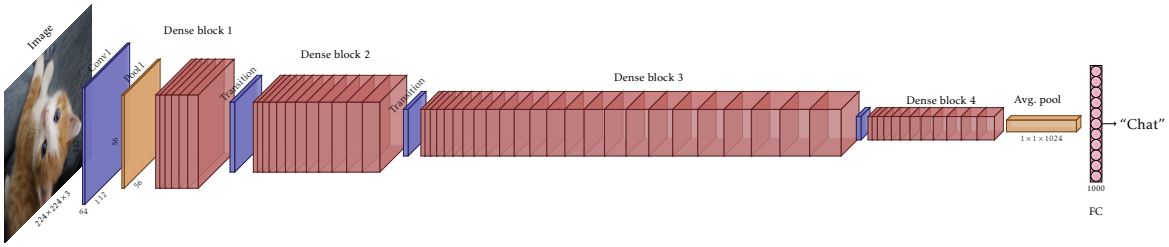


FIGURE 2.18 – Architecture DenseNet-121 [74].

ResNet coûteux en mémoire et peu pratiques sur de grandes images. L'architecture *Inception* sera également améliorée par des connexions résiduelles [165].

L'utilisation de cartes d'activation intermédiaires et leur propagation aux couches supérieures permet une meilleure classification en prenant en compte plusieurs niveaux d'abstraction. En outre, plusieurs travaux suggèrent que ces approches permettent en pratique de combiner plusieurs modèles en un seul, les activations étant en mesure de suivre plusieurs chemins dans la topologie du réseau [174, 73]. Toutefois, les connexions résiduelles ne permettent que d'accéder aux activations de la couche précédente. HUANG et al. [74] ont donc proposé une architecture dite *DenseNet* comportant des connexions denses, construisant un modèle au sein duquel toutes les cartes d'activation issues des couches inférieures sont transmises à toutes les couches supérieures. Pour éviter l'explosion du nombre de paramètres et d'activations, le modèle est divisé en plusieurs blocs denses, comme illustré dans la Figure 2.18. Chaque bloc se détaille comme illustré dans la Figure 2.16. La présence des connexions denses permet au gradient de se propager immédiatement des couches supérieures aux couches inférieures, appliquant ainsi une forme implicite de supervision profonde [95]. Entre deux blocs, une couche convolutive de transition est appliquée pour réduire le nombre de plans et est suivie d'un *max-pooling* pour réduire les dimensions spatiales. Cette architecture obtient comparativement de meilleurs résultats que les modèles ResNet sur la base de validation de l'ILSVRC 2012. Mais, à l'instar des ResNet, si le nombre de paramètres des architectures DenseNet est faible, ces modifications sont coûteuses en espace mémoire nécessaire pour stocker les activations et gradients intermédiaires.

Enfin, CHOLLET [25] introduit les convolutions séparables en profondeur ou *depthwise separable convolutions*. Celles-ci opèrent un filtre par plan du tenseur d'activation et sont recombinées pixel à pixel par une convolution point à point de noyau 1×1. Ainsi, il s'agit d'un cas spécifique de la convolution usuelle dans laquelle chaque plan du tenseur est filtré par un et un seul noyau de convolution, comme illustré dans la Figure 2.14. Ces convolutions sont introduites pour remplacer le module *Inception* dans l'architecture éponyme et en ont amélioré les performances sur les jeux de données ImageNet et JFT, interne à Google. Un avantage notable de ces convolutions est de nécessiter moins de paramètres que la convolution classique. En effet, une convolution $k_1 \times k_2$ opérant sur N_{in} cartes d'activations produisant N_{out} cartes d'activations nécessite $k_1 \times k_2 \times N_{in} \times N_{out}$ paramètres. La convolution séparable en profondeur nécessite $N_{in} \times N_{in} \times k_1 \times k_2$ paramètres pour la première phase puis $N_{in} \times N_{out}$ pour la seconde, soit un total de $N_{in} \times (N_{in} \cdot k_1 \cdot k_2 + N_{out})$, ce qui est avantageux quand

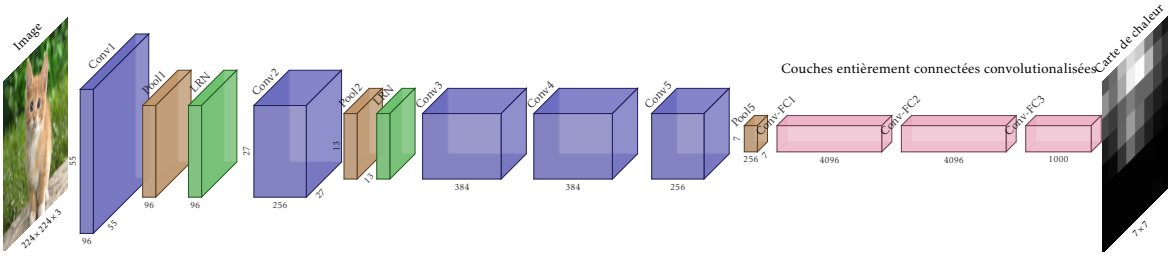


FIGURE 2.19 – AlexNet entièrement convolutif proposé par LONG, SHELHAMER et DARRELL [105].

$N_{out} \geq N_{in}$, ce qui est le cas le plus courant. Ces convolutions séparables sont ainsi efficaces à calculer et de fait populaires pour des applications embarquées en temps réel [70].

Bien que ces succès soient encourageants, il est nécessaire de rappeler que la classification d'images ne donne aucune information particulière de localisation, seulement une information binaire de présence d'un objet dans une image. Les premières approches de localisation ont cherché à calculer des caractéristiques denses sur l'image, associées à des cascades multiéchelles de classifieurs pour détecter des objets dans chaque sous-région de l'image. C'est l'approche utilisée par les descripteurs SIFT [107], les pseudo-Haar de la méthode de VIOLA et JONES [176] ou encore des HOG [35]. Des approches à base de réseaux profonds pour la détection d'objet à partir de classification de sous-régions de l'image ont rapidement vu le jour [53, 103, 54], supplantant les techniques traditionnelles de localisation à partir d'extraction de fenêtres candidates [60, 169]. Le principe reste néanmoins identique. Il s'agit de réaliser une extraction dense de caractéristiques afin d'identifier les régions de l'image susceptibles de contenir un objet d'intérêt. Cependant, ces approches ne répondent pas exactement au problème décrit par Papert en 1966. La question n'est pas uniquement de pouvoir identifier des objets, mais aussi d'en estimer la forme et de les séparer du fond, comme illustré dans la Figure 2.8. Cette tâche est dite de *segmentation sémantique* et correspond à l'association d'une classe d'intérêt non pas à chaque image, mais à chaque pixel de celle-ci.

Plusieurs jeux de données ont été introduits dans la communauté de la vision par ordinateur afin d'évaluer les techniques de segmentation sémantique, notamment sur des scènes de la vie quotidienne, comme PASCAL VOC [44], Microsoft COCO [99] et également sur des scènes de conduite autonome avec des bases de données telles que CamVid [19], Cityscapes [30] ou encore Mapillary Vistas [122]. Les premières approches de segmentation sémantique ont réalisé des classifications à partir de caractéristiques denses calculées sur l'ensemble de l'image, regroupées *a posteriori* par régions homogènes [155, 156]. Les réseaux convolutifs profonds ont également été mis à contribution, en exploitant leur nature convolutive pour réaliser une classification sur chaque pixel de l'image [59, 28] ou pour chaque région [45, 150]. En effet, les cartes de caractéristiques issues des couches convolutives conservent la structure spatiale de l'image. Il est possible de faire correspondre chaque caractéristique à un ou plusieurs pixels de l'image initiale. Cette extraction de caractéristiques dense se prête particulièrement aux problématiques de localisation d'objet et a été rapidement adoptée par la communauté [195]. Nous détaillerons un peu plus en détail certaines de ces approches dans le Chapitre 3. Dans la suite de cette partie, nous nous intéressons plus particulièrement aux réseaux profonds entièrement convolutifs, destinés à la classification dense pixel à pixel. En effet, ceux-ci sont une évolution naturelle des approches d'extraction dense de caractéristique, permettant un apprentissage de bout en bout pour la segmentation sémantique.

2.2.2 Approches entièrement convolutives

La forme moderne des réseaux de neurones entièrement convolutifs, ou *Fully Convolutional Networks* (FCN), pour la segmentation sémantique a été popularisée par Long, Shelhamer et Darrell [105]. Le principe au cœur de cette architecture est de ne manipuler que des réseaux entièrement convolutifs, c'est-à-dire sans couche entièrement connectée (cf. Figure 2.19). Dès lors, les cartes d'activation conservent leurs propriétés spatiales et peuvent être relocalisées sur l'image originale par un simple jeu de sur-échantillonnage, par exemple par interpolation bilinéaire. L'approche choisie par Long, Shelhamer et Darrell [105] consiste à transformer la première couche entièrement connectée en convolution dont le noyau recouvre l'intégralité des cartes d'activation. En effet, ces deux opérations sont mathématiquement équivalentes, mais l'expression sous forme de convolution permet de ne plus se limiter à une taille précise d'images. La première couche entièrement connectée est donc remplacée par une convolution 7×7 et les suivantes par des convolutions de noyaux 1×1. En particulier, cette transformation permet de conserver les poids de l'ensemble du réseau déjà entraîné pour la classification d'images. De fait, ils proposent ainsi d'utiliser les poids de VGG-16 entraîné pour la classification d'objets sur ImageNet en rendant convolutif ses dernières couches afin de générer des prédictions denses à résolution 1 : 32. Ce modèle sert ensuite d'initialisation à un réseau réalisant une segmentation plus fine à résolution 1 : 16 puis 1 : 8 à l'aide d'un décodeur augmentant la résolution des activations. Cette approche permet immédiatement de faire progresser l'état de l'art sur différents jeux de données de segmentation sémantique, notamment PASCAL VOC [44].

Plusieurs axes d'amélioration ont été proposés dans la littérature concernant la segmentation sémantique d'images naturelles à l'aide de FCN. Tout d'abord, en conservant la structure des couches convolutives de VGG-16 [157], Chen et al. [23] proposent d'une part d'utiliser les convolutions à trous pour agrandir le champ réceptif du réseau tout en retirant les couches de *maxpooling*, qui réduisent la résolution spatiale, et d'autre part d'appliquer un champ de Markov conditionnel, ou *Conditional Random Field* (CRF) *a posteriori* pour régulariser les cartes prédites. Dans le même esprit, Yu et Koltun [185] adoptent la convolution à trous (nommée convolution dilatée) pour agréger des cartes d'activation à plusieurs échelles, combinant ainsi l'agrandissement du champ réceptif avec les convolutions parallèles du module *Inception* [163]. Ces modèles réalisent ainsi une extraction dense de caractéristiques sur l'ensemble de l'image, produisant des cartes d'activation à résolution réduite d'un facteur 4 ou 8.

En parallèle, une famille de FCN dérivée des auto-encodeurs convolutifs [190] émerge. Tandis que les modèles de Long, Shelhamer et Darrell [105] consistent en un encodeur profond calqué sur la topologie d'un classifieur usuel, suivi d'un décodeur constitué de peu de couches de déconvolution, ces FCN présentent une architecture symétrique. Il s'agit alors de projeter les cartes d'activation basse résolution issues de l'encodeur dans l'espace des classes à haute résolution, soit par le biais de la convolution transposée [120, 124], soit par un sur-échantillonnage parcimonieux [4]. L'architecture U-Net [141] utilise ainsi des *skip connections* (courts-circuits) pour réinjecter les cartes d'activation des couches de l'encodeur dans la phase de décodage et des convolutions transposées pour reconstituer la résolution originale de l'image. Ces approches utilisent comme encodeur les couches convolutives de CNN préentraînés pour la classification, notamment VGG-16. L'intérêt de ces approches symétriques est de pouvoir générer des prédictions à la même résolution spatiale que l'image d'entrée. En effet, l'encodeur produit des cartes d'activation sous-résolues, d'un facteur 8 pour VGG-16, qui vont être reconstruites à résolution plus élevée par les couches successives du décodeur.

En outre, compte-tenu des performances supérieures obtenues en reconnaissance d'objet par les modèles ResNet et DenseNet, la communauté a également cherché à adapter ces architectures pour la segmentation sémantique. Un verrou majeur de ces approches réside dans leur important coût en mémoire, compte-tenu du nombre important d'activations

intermédiaires à stocker dans des réseaux aussi profonds. L'augmentation des capacités de calcul des GPU aidant, Wu, Shen et Van Den Hengel [182] proposent ainsi une première approche pour les ResNet, qui sera également utilisée pour le modèle DeepLab [23]. Pour réduire la complexité en espace des modèles, les architectures développées suivent l'approche initiale de Long, Shelhamer et Darrell [105] et génèrent des cartes de prédiction à un facteur d'échelle 1 : 4. Récemment, une version entièrement convolutive des DenseNet [79] a également été proposée pour la segmentation sémantique en combinant une approche encodeur-décodeur avec le passage d'activations inspiré de U-Net [141].

Enfin, hormis les travaux sur l'architecture de base des FCN, plusieurs améliorations connexes ont été proposées pour raffiner la qualité des segmentations sémantiques obtenues à partir des différents modèles. La communauté s'est ainsi penchée sur l'utilisation des modèles graphiques structurés pour la régularisation des cartes sémantiques inférées par les FCN. En particulier, les CRF sont reformulés de manière à s'exprimer sous forme d'un réseau récurrent optimisable conjointement avec le FCN [191] ou comme post-traitement [2].

Dans la veine des *skip connections*, le modèle GridNet [48] s'intéresse à des topologies de réseaux non-conventionnelles en proposant une architecture constituée de plusieurs ResNet parallèles dont les activations peuvent évoluer aussi en profondeur que latéralement. C'est également l'approche de Liu et al. [101] dont le décodeur agrège les cartes d'activation en leur permettant de prendre plusieurs chemins parmi le graphe de calcul du réseau convolutif.

Finalement, plusieurs approches multiéchelles ont été proposées. Chen et al. [23] intègrent dans leur modèle DeepLab des prédictions à plusieurs résolutions, interpolées et moyennées en fin de traitement. D'autres approches utilisent des noyaux de convolution de différentes tailles, soit en faisant varier un facteur de dilatation [185], soit en déployant au sein du réseau des modules *Inception* [163, 120, 189]. Peng et al. [134] ont proposé un module global de déconvolution observant l'intégralité de l'image afin de modéliser les relations spatiales à longue distance entre les éléments constitutifs d'une scène. Ils combinent cette technique à un apprentissage résiduel permettant de raffiner les bordures des objets. Dans l'ensemble, les techniques d'inférence multiéchelles pour la segmentation sémantique sont construites en majorité sur un système de convolutions parallèles générant une pyramide de cartes d'activation à différentes résolutions.

En résumé, la segmentation sémantique d'images multimédia est une tâche fréquemment étudiée dans la littérature. Les approches par FCN ont permis d'établir de nouveaux états de l'art sur de nombreux jeux de données : Microsoft COCO [99], PASCAL VOC [44], Cityscapes [30] ou ADE20k [192]. Toutefois, ceux-ci se focalisent sur la compréhension de scènes de la vie quotidienne : images d'intérieurs ou de conduite urbaine contenant de nombreux objets observés par une multitude de points de vue avec occlusions, acquises par des appareils photos ou des caméras du commerce. La contribution centrale de cette thèse consiste ainsi à comprendre dans quelle mesure les images d'observation de la Terre peuvent bénéficier des connaissances et techniques mises en place sur ces données multimédia.



(a) Image de la Caroline du Nord (États-Unis). Le codage couleur correspond à l'élévation topographique.
Crédits images : Cintos (domaine public, Wikimédia Commons)

(b) Composition RVB d'une image multispectrale Sentinel-2 sur l'île de Viti Levu (Fiji).
Crédits images : données Copernicus Sentinel, traitées par l'ESA (CC BY-SA 3.0 IGO)

(c) Image Sentinel-1 de la barrière de glace de Dotson (Antarctique).
Crédits images : données Copernicus Sentinel, traitées par A. Hogg/CPOM

Figure 2.20 – L'observation de la Terre implique une grande variété de capteurs dotés de spécificités qui leur sont propres.

2.3 Apprentissage pour le traitement d'images de télédétection

L'interprétation d'images de télédétection mobilise des fonctions cognitives similaires à celles utilisées pour la compréhension de photographies de la vie quotidienne. Les outils de traitement d'images et de vision artificielle sont ainsi très largement mis en œuvre pour l'aide à la photo-interprétation. Cependant, l'observation de la Terre fait appel à des capteurs et à des points de vue très spécifiques. Ainsi, la cartographie automatisée d'images aériennes et satellitaires ne se réduit pas à une branche de la vision artificielle. Il s'agit de l'intersection entre celle-ci et la télédétection pour l'observation de la Terre, recouvrant aussi bien des aspects d'apprentissage automatique pour la vision que du traitement des signaux spécifiques aux capteurs aéroportés et satellites, souvent très éloignés des appareils photos du commerce.

2.3.1 Différents types d'imagerie

La Figure 2.20 présente quelques capteurs illustrant la variété des appareils d'imagerie de télédétection pour l'observation de la Terre. Si les acquisitions aéroportées peuvent s'effectuer en couleurs RVB à l'aide d'appareils photos classiques, les acquisitions satellitaires utilisent bien souvent des capteurs sophistiqués dotés de capacités particulières, comme transmettre une information spectrale riche, percer la couverture nuageuse ou mesurer des propriétés physiques de la surface de la Terre.

Par exemple, l'imagerie infrarouge est souvent utilisée en complément des acquisitions couleur. Il est courant, y compris en aéroporté, d'imager à la fois le domaine visible et le proche infrarouge, situé entre 780 nm et 2500 nm. En effet, la végétation y a une réponse amplifiée par la présence de chlorophylle ce qui rend ce signal très informatif. Dans l'infrarouge moyen et lointain, il est également possible de mesurer indirectement la température par le biais des radiations lumineuses émises selon la loi de Wien. Ces caméras thermiques sont particulièrement utiles dans l'espace, où l'influence de la chaleur naturelle terrestre est peu présente.

La Figure 2.21 illustre une caméra multispectrale ou superspectrale qui, selon ce même principe, permet d'imager une scène dans plusieurs bandes de longueurs d'onde plus ou moins larges pouvant se trouver aussi bien dans le domaine visible que dans l'infrarouge ou l'ultraviolet. Cela conduit à des images à plusieurs canaux, généralement une dizaine,

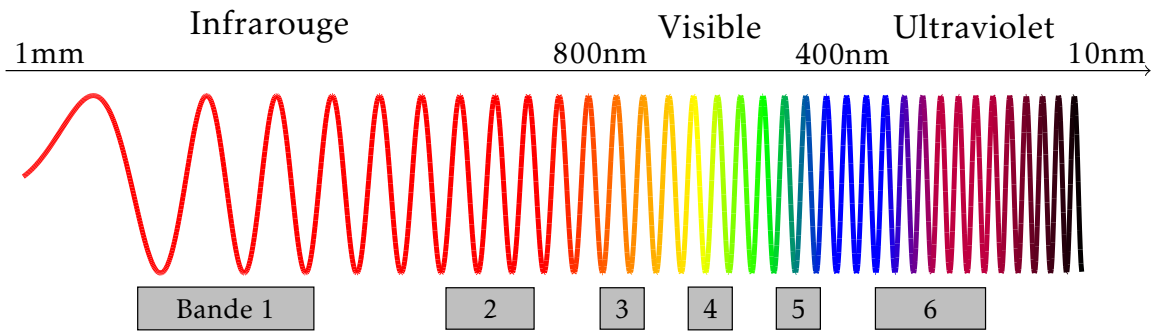


FIGURE 2.21 – Un capteur multispectral acquiert plusieurs bandes spectrales larges réparties sur le spectre lumineux infrarouge, visible et parfois ultraviolet.

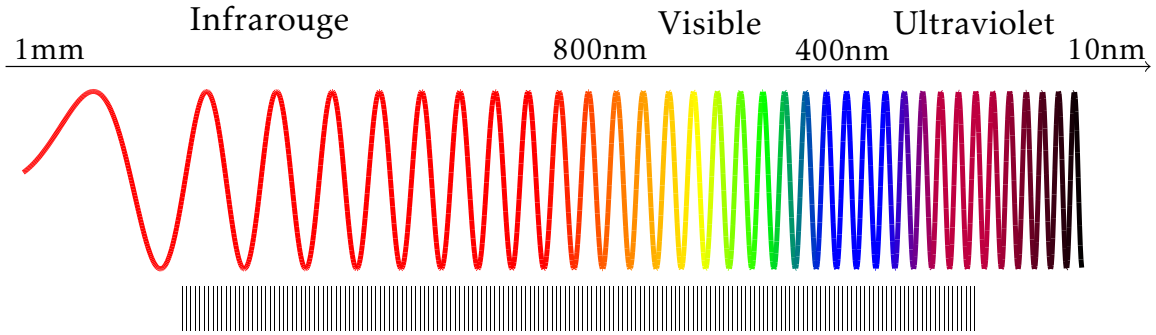


FIGURE 2.22 – Un capteur hyperspectral acquiert de nombreuses bandes spectrales étroites régulièrement réparties sur sa plage d'acquisition.

qui ne sont donc pas directement visualisables par l'œil humain. Il est toutefois possible de reconstituer une image en couleurs naturelles en recomposant une image RVB à partir des valeurs contenues dans les canaux correspondant aux longueurs d'onde du rouge, du vert et du bleu. Les acquisitions multispectrales peuvent avoir des résolutions différentes en fonction des canaux. Les satellites Sentinel-2A et Sentinel-2B produisent par exemple des images à une résolution au sol de 10 m/px dans le domaine visible, mais certaines bandes, notamment dans l'infrarouge, ont des résolutions de 20 m/px ou 60 m/px. Les acquisitions couleurs satellitaires les mieux résolues spatialement ont une résolution au sol de l'ordre de 30 cm/px, tandis que les images aéroportées peuvent aller jusqu'à une résolution de 5 cm/px. Dans le cas des acquisitions satellitaires, il est courant de réaliser en simultané une mesure multispectrale et une acquisition panchromatique, c'est-à-dire qui ne distingue pas les couleurs et produit une image en noir et blanc, à résolution supérieure. Ainsi, la constellation de satellites Pléiades réalise à la fois une acquisition panchromatique à résolution 70 cm/px, ré-échantillonnée à 50 cm/px et multispectrale à 2,8 m/px rééchantillonnée à 2 m/px.

L'imagerie hyperspectrale consiste à effectuer des acquisitions sur de nombreuses bandes étroites, toutes de même taille, afin de balayer de façon discrète l'ensemble du spectre lumineux réfléchi, comme illustré par la Figure 2.22. En fonction de la résolution spectrale – souvent de l'ordre de 10 nm – et de la largeur du spectre considéré, le nombre de bandes peut varier de quelques dizaines à plusieurs centaines. L'intérêt de ces caméras réside dans la possibilité de reconstituer la courbe de l'intensité lumineuse réfléchie en fonction de la longueur d'onde pour chaque pixel. En effet, tous les matériaux réfléchissent différemment la lumière du Soleil en fonction de leur albédo. Cette information permet donc de caractériser finement la composition physique des objets observés. En contrepartie, la résolution spatiale des caméras hyperspectrales est nettement plus faible que celle des autres appareils optiques. Les acquisitions hyperspectrales présentent de fait une résolution au sol d'environ 1 m/px en aérien et 30 m/px en satellite.

Notons que l'ensemble des capteurs optiques présentés ci-dessus sont passifs ; ils ne

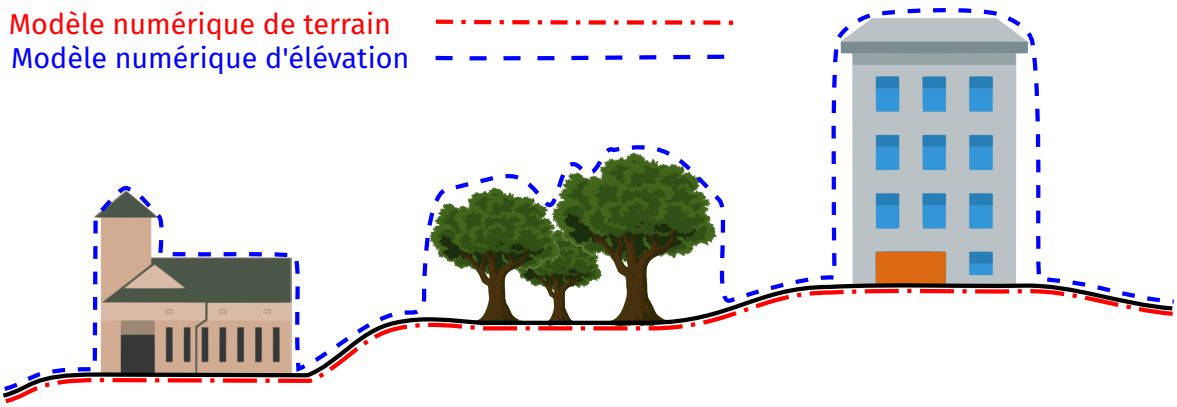


Figure 2.23 – Schéma représentatif de la différence entre MNT et MNE.

reçoivent que l'énergie lumineuse réfléchie ou émise par le corps observé. Par conséquent, ces capteurs sont sensibles aux variations d'illumination et aux effets météorologiques, les nuages pouvant notamment les rendre entièrement inopérationnels. De nombreux autres satellites sont munis de capteurs actifs qui émettent un signal dont ils mesurent le retour. C'est notamment le cas des satellites radar, et en particulier radar à synthèse d'ouverture (en anglais *Synthetic Aperture Radar*) (SAR), qui envoient une ou plusieurs ondes électromagnétiques et utilisent la mesure de l'onde réfléchie pour extraire des paramètres physiques de la zone observée. Le SAR permet ainsi de percer la couverture nuageuse, mais ne produit pas des images à proprement parler.

Le *Light Detection And Ranging* (Lidar) est un autre capteur actif, qui émet une impulsion laser dont il mesure l'écho. La position du maximum d'amplitude permet de déterminer le temps d'aller-retour du rayon lumineux et donc de mesurer la distance parcourue par les photons. Ces capteurs sont très utilisés en télédétection comme en robotique afin d'effectuer des relevés topographiques ou des reconstructions 3D. La nature ponctuelle de la mesure laser ne permet cependant de construire que des nuages de points faiblement denses. Dans le cas du Lidar satellitaire, les mesures se font tous les 20 m, et tous les 10 cm dans le cas de l'aéroporté. Une fois le nuage de points construit, il est possible d'en extraire un maillage qui modélise la topographie de la surface observée. Celui-ci peut alors être rastérisé, c'est-à-dire projeté sur un plan, pour obtenir un Modèle Numérique de Terrain (MNT) ou un Modèle Numérique d'Élévation (MNE) en fonction de la résolution. Le MNE se distingue du Modèle Numérique de Terrain (MNT) en ce qu'il prend en compte les objets surélevés par-dessus la surface topographique, un exemple étant exposé dans la Figure 2.23. La différence entre ces deux valeurs est appelée Modèle Numérique de Hauteur (MNH) et correspond à la hauteur normalisée des points au-dessus du sol.

Les travaux présentés dans ce manuscrit portent principalement sur l'utilisation de données optiques pour la cartographie automatisée. Nous nous autoriserons néanmoins à faire appel à des données ancillaires, qu'elles soient dérivées d'acquisitions Lidar ou provenant de systèmes d'information géographiques (SIG).

2.3.2 Apprentissage et images de télédétection

Comme nous venons de le voir, les images de télédétection peuvent se présenter sous des formes variées : caméra multispectrales, imagerie hyperspectrales, SAR, Lidar… De nombreuses techniques d'apprentissage automatique ont été mises en œuvre pour extraire de l'information de ces données sans intervention humaine.

Extraction de caractéristiques

Une fois les données acquises et mises en forme, il est nécessaire de choisir une méthode de représentation adaptée à la classification. Plus précisément, il s'agit de décider d'un espace de représentation dans lequel projeter les données, de manière à ce qu'il soit aisé de partitionner l'espace afin d'y séparer les différentes classes. Nous présentons ci-dessous plusieurs approches couramment mises en œuvre.

Une première méthode consiste à directement utiliser les données brutes, éventuellement normalisées. Le classifieur opère directement sur les valeurs de luminance ou de réflectance des pixels. Cependant, une image RVB de dimensions 128×128 est alors décrite par 128×128×3 = 49152 scalaires, ce qui est intraitable pour la plupart des modèles statistiques usuels. Cela impose alors de traiter les pixels individuellement ou de découper l'image en régions de petite taille. De telles approches sont fréquentes pour le traitement de données hyperspectrales [46, 62] et multispectrales, y compris infrarouge-rouge-vert-bleu (IRRVB) [39].

En effet, il est possible de combiner les données brutes à des caractéristiques expertes, comme les moments statistiques ou le gradient du signal. Dans le cas des images Lidar, l'écart local à la hauteur moyenne est un indicateur souvent adopté pour discriminer différents types d'objets [61, 97, 90], et l'entropie locale est une caractéristique classique de l'imagerie SAR [5].

En outre, les capteurs multispectraux et SAR présentent des propriétés physiques dont la connaissance *a priori* est exploitable. Des rapports de réflectance dans différentes longueurs d'onde peuvent permettre de caractériser certaines surfaces, comme le *Normalized Difference Vegetation Index* (NDVI) [143] pour la végétation et le *Normalized Difference Water Index* (NDWI) [183] pour l'eau. Ces indices sont facilement interprétables mais savoir lesquels utiliser demande une connaissance experte des phénomènes étudiés (il est impossible de détecter un phénomène que l'on ne saurait pas caractériser physiquement, au moins partiellement) et un effort systématique d'ingénierie, puisque ces indices dépendent du problème considéré.

Comme pour le traitement d'images multimédia, il peut être intéressant de rechercher des caractéristiques génériques. Les histogrammes de couleurs peuvent par exemple s'appliquer à des images multispectrales ou hyperspectrales de la même façon qu'ils s'appliquent aux images RVB. De tels histogrammes sont invariants aux rotations et aux translations locales, ainsi qu'aux changements d'échelle. Toutefois, ils sont fortement influencés par les changements radiométriques induits par l'environnement, comme la variation de la luminosité extérieure. Qui plus est, si la discrétisation des valeurs dans l'histogramme ajoute une robustesse au bruit, elle diminue la précision des valeurs numériques et risque de faire disparaître des différences subtiles entre spectres. Dans le cas de l'hyperspectral, deux plastiques peuvent présenter des profils spectraux très similaires, ne différant que par la position de leurs pics d'absorption. Une quantification trop importante peut alors faire disparaître ces pics et donc l'information discriminante. D'autres histogrammes usuels, comme les HOG [35] s'appliquent également aux images de télédétection.

Enfin, les profils morphologiques ont également été fréquemment étudiés pour la classification d'images de télédétection. En particulier, les travaux de Benediktsson, Pesaresi et Árnason [6] ont introduit ces descripteurs, obtenus par l'application successive d'opérations morphologiques d'érosion et de dilatation. Ces caractéristiques ont l'avantage de renseigner sur l'appartenance d'un pixel à des structures spatiales à différentes échelles sous la forme de profils d'attributs [36].

En pratique, il est courant d'utiliser une combinaison de ces différentes caractéristiques, en les calculant par exemple sur une pyramide d'images multiéchelle. Une fois les vecteurs de caractéristiques générés, de nombreux modèles statistiques peuvent alors être appliqués.

Modèles statistiques usuels

Une fois le vecteur de caractéristiques extrait pour un échantillon, celui-ci sert alors d'entrée à un classifieur. Le classifieur est un modèle statistique de décision pouvant prendre plusieurs formes. Cette partie ne traite que des classifieurs sans apprentissage de représentation, excluant par conséquent les réseaux de neurones profonds qui seront discutés plus tard.

La littérature en apprentissage automatique pour la télédétection a longtemps plébiscité les arbres de décision sous forme de forêts aléatoires [17] et les SVM [13, 31].

Les arbres de décision [18] forment un ensemble de modèles statistiques représentant les variables sous forme de nœud intérieur, chaque arête correspondant à un ensemble de valeurs possibles pour la variable associée au nœud. L'ensemble des arêtes partant d'un nœud donné couvre l'ensemble des valeurs que peut prendre la variable qui lui est associée. Durant la phase d'apprentissage, l'arbre est construit par partitionnement récursif, divisant l'ensemble des données en fonction d'une première variable, puis d'une seconde et ainsi de suite jusqu'à ce que l'ajout de variable n'améliore plus la prédiction, ou que tous les sous-ensembles aboutissent au même résultat.

Les arbres de décision sont couramment utilisés sous forme de forêts aléatoires [17]. Une *Random Forest*, ou forêt aléatoire, (RF) est en réalité un ensemble d'arbres de décisions construits à partir de sous-ensembles aléatoires des variables d'entrée. Chaque arbre réalise sa prédiction indépendamment des autres, et la prédiction finale est celle ayant obtenu le plus de votes. Réaliser un tel apprentissage par ensemble permet d'obtenir un classifieur dont la variance est plus faible que chaque arbre individuel. Les arbres de décision ont l'avantage d'être simples à interpréter, puisqu'une décision (un nœud) est associée à un test sur une caractéristique précise. Les forêts aléatoires ont ainsi été abondamment utilisées en télédétection pour la cartographie d'occupation des sols à partir d'images Landsat [128] et la prédiction météorologique [88]. Les ensembles d'arbres de décision ont également été mis en œuvre sur le principe du *gradient boosting*, permettant d'exploiter un ensemble de modèles de prédiction faibles pour les combiner et renforcer leur pouvoir prédictif [50]. Ce type de classifieurs est plus rare en télédétection mais se retrouve néanmoins dans certaines applications [89].

Les SVM [13, 31] sont des classifieurs partitionnant l'espace de telle sorte que la distance entre la frontière et l'échantillon le plus proche (la marge) soit maximale. La frontière se représente sous la forme d'un hyperplan dans l'espace des données d'entrée dans le cas du noyau linéaire, ou d'un hyperplan dans un espace de représentation de grande dimension (possiblement infinie) en utilisant l'astuce du noyau [13].

Si la dimension des données d'entrée est de grande taille, le calcul exact des hyperplans à marge maximale n'est pas nécessairement réalisable en temps raisonnable. Dans ce cas, il est possible de faire appel à des algorithmes d'approximation reposant sur une optimisation par descente de gradient [15].

Les SVM ont trouvé de nombreuses applications en télédétection, notamment pour l'occupation des sols à partir d'images multispectrales [129] et hyperspectrales [114].

Enfin, on retrouve également des réseaux de neurones de type perceptron multicouche dans la littérature pour le traitement d'images multispectrales [7] et hyperspectrales [57].

Caractéristiques spatiales et caractéristiques spectrales

Comme nous l'avons vu, les caractéristiques expertes de la littérature en télédétection s'intéressent essentiellement à l'information radiométrique. La plupart des méthodes produisent ainsi une classification pixel à pixel, c'est-à-dire que les classifieurs ne réalisent qu'une seule prédiction à la fois, généralement pour le pixel central d'une région considérée.

Si cette approche garantit que les prédictions auront une résolution identique à celle de la donnée, cela ne permet pas de modéliser les relations spatiales entre objets. Compte-tenu de

l'augmentation continue de la résolution des capteurs, les objets d'intérêt s'étendent souvent sur plusieurs pixels et présentent des propriétés géométriques particulières de connexité et de convexité. Ainsi, un pixel particulier soumis à un bruit extrême (suite à une défaillance du capteur ou à un matériau particulier) peut être mal classifié s'il est considéré isolément. Un modèle considérant l'intégralité de son voisinage pourrait contourner cette erreur en se basant sur des critères d'homogénéité, par exemple. Ainsi la classification pixellique tend à produire des cartes exhibant un bruit poivre-et-sel, qui sont ensuite régularisées *a posteri* par des modèles graphiques de l'état de l'art, comme les CRF.

En réponse, des approches de classification par *patch* sont apparues pour tirer parti du contexte spatial des objets. Ces techniques parcourent l'image par une fenêtre glissante et classifient pour chaque voisinage carré le pixel central. Ces approches ont connu un succès considérable grâce aux progrès en classification d'images apportés par les CNN. Initialement, la communauté a introduit des descripteurs experts à la fois spatiaux et spectraux [46], mais les représentations spatiales-spectrales automatiquement apprises par les réseaux profonds se montrent bien souvent plus performantes [123, 24]. Plusieurs travaux déploient ainsi des CNN via une fenêtre glissante pour la détection de bâtiments [172] et l'étude de l'occupation des sols [130].

Néanmoins, le nombre de pixels croissant quadratiquement avec la taille de l'image, ces approches passent difficilement à l'échelle, notamment en imagerie haute résolution (HR), très haute résolution (THR) et extrêmement haute résolution (EHR). Il est en effet inenvisageable de calculer une prédiction par pixel sur des images en comportant plusieurs millions, voire plusieurs milliards. L'alternative consiste à diminuer le nombre de passes d'inférence nécessaires en réalisant non pas une prédiction pour un unique pixel, mais pour une région toute entière.

L'approche de classification par régions consiste ainsi à regrouper ensemble des pixels similaires en régions homogènes. Le critère de similarité utilisé pour fusionner les pixels dépend à la fois de leurs valeurs et de leur position. Le classifieur réalise ensuite une inférence unique pour l'ensemble des pixels d'une même région, en faisant l'hypothèse que des pixels spatialement et spectralement proches possèdent la même sémantique, c'est-à-dire que l'on puisse les associer à la même classe d'intérêt. Dans ce cas, il suffit alors d'extraire des caractéristiques pour chaque région. Ainsi, dans le cas d'une image de dimensions 1500×1500 segmentée en 20000 régions, il devient possible de cartographier l'image en classifiant 20000 régions plutôt que 2250000 pixels.

De nombreux algorithmes de segmentation ont été proposés, aussi bien dans la communauté de la télédétection que dans la communauté de la vision par ordinateur. Ces algorithmes partitionnent l'ensemble des pixels de façon non-supervisée. Une fois ce partitionnement effectué, il est possible d'extraire des attributs pour chaque région et d'entraîner un classifieur de la façon habituelle.

Réaliser une classification par régions permet de significativement réduire la complexité calculatoire de la cartographie. Lorsque l'image augmente de résolution spatiale, la plupart des régions conservent leur homogénéité et il est ainsi avantageux de les conserver regroupées : l'augmentation de la résolution n'implique pas obligatoirement une évolution quadratique du nombre de régions. Depuis les premiers travaux de Mnih [117] utilisant les CNN pour l'extraction de routes et de bâtiments dans des images aériennes à partir d'imagettes, ces approches ont ainsi été utilisées avec succès sur de nombreux jeux de données THR [86, 173]. Notons par ailleurs que traiter l'image par une fenêtre glissante ou même par une grille de pixels correspond à des cas particuliers de classification par région.

Dans les cas des profils morphologiques, ces approches par présegmentation sont d'autant plus intéressantes que les opérateurs morphologiques sont coûteux à calculer. Une alternative classique est d'opérer sur une segmentation multiéchelle hiérarchique de l'image, représentée sous forme d'arbre. Les opérations de filtrage morphologique se traduisent en opérations d'élagage de l'arbre, rapides à calculer. De nombreuses approches de construction d'arbres

existent [14], permettant de construire des profils d'attributs, c'est-à-dire des caractéristiques multiéchelles étendant les propriétés des profils morphologiques. Ces approches ont longtemps représenté l'état de l'art en cartographie automatisée [135] et des modèles statistiques spécifiques ont été développés pour en tirer parti [33].

En parallèle de cette thèse, les approches utilisant les réseaux entièrement convolutifs (FCN) pour la segmentation sémantique d'images de télédétection se sont nettement popularisées. En effet, les FCN infèrent une prédiction pixellique pour l'ensemble de l'image en une seule passe, s'affranchissant ainsi du problème de la classification par *patch*. Cela réduit drastiquement les temps de calcul, sans pour autant nécessiter une présegmentation non supervisée. Les premières applications de FCN sur des données aériennes optiques apparaissent en 2015 chez Paisitkriangkrai et al. [127] et Sherrah [154] en se basant sur les architectures initiales de Long, Shelhamer et Darrell [105]. Les modèles encodeur-décodeurs symétriques suivent rapidement [177, 3] et sont le sujet de plusieurs travaux dérivés, comme la conception d'un CRF [104] pour la fusion de données ou la régularisation des bordures par contrainte explicite [112]. Si les images aériennes EHR sont naturellement les premières candidates, les FCN sont également mis en œuvre pour la cartographie d'images satellites [51].

De nouveaux jeux de données comme l'*Inria Aerial Image Labeling Dataset* fournissent un terrain de jeu propice aux réseaux entièrement convolutifs, qui dominent très largement les classements [71]. Dans l'ensemble, les méthodes par apprentissage profond utilisant les FCN se sont imposées en quelques années comme le nouvel état de l'art sur de nombreuses tâches d'interprétation d'images de télédétection [102]. L'apparition de nombreux jeux de données annotés de télédétection, détaillés en Annexe A, a notamment permis d'entraîner des modèles profonds sur une large gamme d'applications.

Références

[1] David H. Ackley, Geoffrey E. Hinton et Terrence J. Sejnowski. « A Learning Algorithm for Boltzmann Machines ». Dans : *Cognitive Science* 9.1 (1er jan. 1985), p. 147-169. ISSN : 0364-0213. DOI : 10.1016/S0364-0213(85)80012-4. URL : http://www.sciencedirect.com/science/article/pii/S0364021385800124 (cf. p. 12).

[2] Anurag Arnab et al. « Higher Order Conditional Random Fields in Deep Neural Networks ». Dans : *Computer Vision – ECCV 2016*. Sous la dir. de Bastian Leibe et al. Lecture Notes in Computer Science. Springer International Publishing, 2016, p. 524-540. ISBN : 978-3-319-46475-6 (cf. p. 36).

[3] Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « Semantic Segmentation of Earth Observation Data Using Multimodal and Multi-Scale Deep Networks ». Dans : *Computer Vision – ACCV 2016*. Springer, Cham, 20 nov. 2016, p. 180-196. DOI : 10.1007/978-3-319-54181-5_12 (cf. p. 43).

[4] Vijay Badrinarayanan, Alex Kendall et Roberto Cipolla. « SegNet : A Deep Convolutional Encoder-Decoder Architecture for Scene Segmentation ». Dans : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.12 (déc. 2017), p. 2481-2495. ISSN : 0162-8828. DOI : 10.1109/TPAMI.2016.2644615 (cf. p. 35).

[5] David G. Barber et Ellsworth LeDrew. « SAR Sea Ice Discrimination Using Texture Statistics : A Multivariate Approach ». Dans : *Photogrammetric Engineering and Remote Sensing* 57 (1er avr. 1991) (cf. p. 40).

[6] Jón Atli Benediktsson, Martino Pesaresi et Kolbeinn Árnason. « Classification and Feature Extraction for Remote Sensing Images from Urban Areas Based on Morphological Transformations ». Dans : *IEEE Transactions on Geoscience and Remote Sensing* 41.9 (sept. 2003), p. 1940-1949. ISSN : 0196-2892. DOI : 10.1109/TGRS.2003.814625 (cf. p. 40).

[7] Jón Atli Benediktsson, Philip H. Swain et Okan K. Ersoy. « Neural Network Approaches Versus Statistical Methods In Classification Of Multisource Remote Sensing Data ». Dans : *IEEE Transactions on Geoscience and Remote Sensing* 28.4 (juil. 1990), p. 540-552. ISSN : 0196-2892. DOI : 10.1109/TGRS.1990.572944 (cf. p. 41).

[8] Yoshua Bengio. « Learning Deep Architectures for AI ». Dans : *Foundations and trends® in Machine Learning* 2.1 (2009), p. 1-127 (cf. p. 12).

[9] Yoshua Bengio. « Practical Recommendations for Gradient-Based Training of Deep Architectures ». Dans : *Neural Networks : Tricks of the Trade*. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, 2012, p. 437-478. ISBN : 978-3-642-35288-1 978-3-642-35289-8. DOI : 10.1007/978-3-642-35289-8_26. URL : https://link.springer.com/chapter/10.1007/978-3-642-35289-8_26 (cf. p. 20).

[10] Yoshua Bengio et al. « Greedy Layer-Wise Training of Deep Networks ». Dans : *Advances in Neural Information Processing Systems 19*. Sous la dir. de B. Schölkopf, J. C. Platt et T. Hoffman. MIT Press, 2007, p. 153-160. URL : http://papers.nips.cc/paper/3048-greedy-layer-wise-training-of-deep-networks.pdf (cf. p. 12, 19).

[11] Monica Bianchini et Franco Scarselli. « On the Complexity of Neural Network Classifiers : A Comparison Between Shallow and Deep Architectures ». Dans : *IEEE Transactions on Neural Networks and Learning Systems* 25.8 (août 2014), p. 1553-1565. ISSN : 2162-237X. DOI : 10.1109/TNNLS.2013.2293637 (cf. p. 16).

[12] Margaret Boden. *Mind as Machine : A History of Cognitive Science*. Oxford ; New York : OUP Oxford, 26 juin 2008. 1756 p. ISBN : 978-0-19-954316-8 (cf. p. 28).

[13] Bernhard E. Boser, Isabelle M. Guyon et Vladimir Vapnik. « A Training Algorithm for Optimal Margin Classifiers ». Dans : *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*. COLT '92. New York, NY, USA : ACM, 1992, p. 144-152. ISBN : 978-0-89791-497-0. DOI : 10.1145/130385.130401. URL : http://doi.acm.org/10.1145/130385.130401 (cf. p. 41).

[14] Petra Bosilj et al. « Partition and Inclusion Hierarchies of Images : A Comprehensive Survey ». Dans : *Journal of Imaging* 4.2 (1er fév. 2018), p. 33. DOI : 10.3390/jimaging4020033. URL : http://www.mdpi.com/2313-433X/4/2/33 (cf. p. 43).

[15] Léon Bottou. « Large-Scale Machine Learning with Stochastic Gradient Descent ». Dans : *In COMPSTAT*. 2010 (cf. p. 41).

[16] Léon Bottou. « Stochastic Gradient Descent Tricks ». Dans : *Neural Networks : Tricks of the Trade*. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, 2012, p. 421-436. ISBN : 978-3-642-35288-1 978-3-642-35289-8. DOI : 10.1007/978-3-642-35289-8_25. URL : https://link.springer.com/chapter/10.1007/978-3-642-35289-8_25 (cf. p. 19, 20).

[17] Leo Breiman. « Random Forests ». Dans : *Machine Learning* 45.1 (1er oct. 2001), p. 5-32. ISSN : 0885-6125, 1573-0565. DOI : 10.1023/A:1010933404324. URL : https://link.springer.com/article/10.1023/A:1010933404324 (cf. p. 41).

[18] Leo Breiman. *Classification and Regression Trees*. Routledge, 2017 (cf. p. 41).

[19] Gabriel J. Brostow, Julien Fauqueur et Roberto Cipolla. « Semantic Object Classes in Video : A High-Definition Ground Truth Database ». Dans : *Pattern Recognition Letters*. Video-based Object and Event Analysis 30.2 (15 jan. 2009), p. 88-97. ISSN : 0167-8655. DOI : 10.1016/j.patrec.2008.04.005. URL : http://www.sciencedirect.com/science/article/pii/S0167865508001220 (cf. p. 34).

[20] Augustin Louis Cauchy. *Comptes rendus hebdomadaires des séances de l'Académie des sciences*. ark :/12148/bpt6k2982c. Paris : Gauthier-Villars, juil. 1847. URL : http://gallica.bnf.fr/ark:/12148/bpt6k2982c (cf. p. 16).

[21] Ken Chatfield et al. « Return of the Devil in the Details : Delving Deep into Convolutional Nets ». Dans : *Proceedings of the British Machine Vision Conference*. British Machine Vision Conference (BMVC). British Machine Vision Association, 2014, p. 6.1-6.12. isbn : 978-1-901725-52-0. doi : 10.5244/C.28.6. url : http://www.bmva.org/bmvc/2014/papers/paper054/index.html (cf. p. 31).

[22] Kumar Chellapilla, Sidd Puri et Patrice Simard. « High Performance Convolutional Neural Networks for Document Processing ». Dans : *Tenth International Workshop on Frontiers in Handwriting Recognition*. Suvisoft, 2006 (cf. p. 12).

[23] Liang-Chieh Chen et al. « DeepLab : Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs ». Dans : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40.4 (avr. 2018), p. 834-848. issn : 0162-8828. doi : 10.1109/TPAMI.2017.2699184 (cf. p. 35, 36).

[24] Yushi Chen et al. « Deep Feature Extraction and Classification of Hyperspectral Images Based on Convolutional Neural Networks ». Dans : *IEEE Transactions on Geoscience and Remote Sensing* 54.10 (oct. 2016), p. 6232-6251. issn : 0196-2892. doi : 10.1109/TGRS.2016.2584107 (cf. p. 42).

[25] François Chollet. « Xception : Deep Learning with Depthwise Separable Convolutions ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, United States, juil. 2017, p. 1800-1807. doi : 10.1109/CVPR.2017.195 (cf. p. 32, 33).

[26] Dan C. Cireşan, Ueli Meier et Jürgen Schmidhuber. « Multi-Column Deep Neural Networks for Image Classification ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Washington, DC, United States, 2012, p. 3642-3649. isbn : 978-1-4673-1226-4. url : http://dl.acm.org/citation.cfm?id=2354409.2354694 (cf. p. 12).

[27] Dan C. Cireşan et al. « Deep Big Simple Neural Nets Excel on Handwritten Digit Recognition ». Dans : *Neural Computation* 22.12 (déc. 2010), p. 3207-3220. issn : 0899-7667, 1530-888X. doi : 10.1162/NECO_a_00052. arXiv : 1003.0358 (cf. p. 12).

[28] Dan C. Cireşan et al. « Deep Neural Networks Segment Neuronal Membranes in Electron Microscopy Images ». Dans : *Advances in Neural Information Processing Systems*. 2012, p. 2843-2851 (cf. p. 34).

[29] Djork-Arné Clevert, Thomas Unterthiner et Sepp Hochreiter. « Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs) ». Dans : *Proceedings of the International Conference on Learning Representations (ICLR)*. 23 nov. 2015. url : http://arxiv.org/abs/1511.07289 (cf. p. 15).

[30] Marius Cordts et al. « The Cityscapes Dataset for Semantic Urban Scene Understanding ». Dans : *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, United States, juin 2016, p. 3213-3223. doi : 10.1109/CVPR.2016.350 (cf. p. 34, 36).

[31] Corinna Cortes et Vladimir Vapnik. « Support-Vector Networks ». Dans : *Machine Learning* 20.3 (1er sept. 1995), p. 273-297. issn : 0885-6125, 1573-0565. doi : 10.1007/BF00994018. url : https://link.springer.com/article/10.1007/BF00994018 (cf. p. 41).

[32] Daniel Crevier. *AI : The Tumultuous History of the Search for Artificial Intelligence*. New York, NY, USA : Basic Books, Inc., 1993. isbn : 978-0-465-02997-6 (cf. p. 28).

[33] Yanwei Cui, Laetitia Chapel et Sébastien Lefèvre. « Scalable Bag of Subpaths Kernel for Learning on Hierarchical Image Representations and Multi-Source Remote Sensing Data Classification ». Dans : *Remote Sensing* 9.3 (24 fév. 2017), p. 196. doi : 10.3390/rs9030196. url : http://www.mdpi.com/2072-4292/9/3/196 (cf. p. 43).

[34] George Cybenko. « Approximation by Superpositions of a Sigmoidal Function ». Dans : *Mathematics of Control, Signals and Systems* 2.4 (1er déc. 1989), p. 303-314. issn : 0932-4194, 1435-568X. doi : 10.1007/BF02551274. url : https://link.springer.com/article/10.1007/BF02551274 (cf. p. 11, 15).

[35] Navneet Dalal et Bill Triggs. « Histograms of Oriented Gradients for Human Detection ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Juin 2005, p. 886-893. doi : 10.1109/CVPR.2005.177 (cf. p. 12, 22, 34, 40).

[36] Mauro Dalla Mura et al. « Morphological Attribute Profiles for the Analysis of Very High Resolution Images ». Dans : *IEEE Transactions on Geoscience and Remote Sensing* 48.10 (oct. 2010), p. 3747-3762. issn : 0196-2892. doi : 10.1109/TGRS.2010.2048116 (cf. p. 40).

[37] Ingrid Daubechies. *Ten Lectures on Wavelets*. Philadelphia, PA, USA : Society for Industrial and Applied Mathematics, 1992. isbn : 978-0-89871-274-2 (cf. p. 22).

[38] Guillaume de L'Hôpital. *Analyse des infiniment petits, pour l'intelligence des lignes courbes*. Paris : Montalant, 1716. 227 p. url : http://archive.org/details/infinimentpetits1716lhos00uoft (cf. p. 17).

[39] Clément Dechesne et al. « Semantic Segmentation of Forest Stands of Pure Species Combining Airborne Lidar Data and Very High Resolution Multispectral Imagery ». Dans : *ISPRS Journal of Photogrammetry and Remote Sensing* 126 (1er avr. 2017), p. 129-145. issn : 0924-2716. doi : 10.1016/j.isprsjprs.2017.02.011. url : http://www.sciencedirect.com/science/article/pii/S0924271616302763 (cf. p. 40).

[40] Jia Deng et al. « ImageNet : A Large-Scale Hierarchical Image Database ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Juin 2009, p. 248-255. doi : 10.1109/CVPR.2009.5206848 (cf. p. 12, 28, 30).

[41] Chao Dong, Chen Change Loy et Xiaoou Tang. « Accelerating the Super-Resolution Convolutional Neural Network ». Dans : *Computer Vision – ECCV 2016*. European Conference on Computer Vision. Lecture Notes in Computer Science. Springer, Cham, 8 oct. 2016, p. 391-407. isbn : 978-3-319-46474-9 978-3-319-46475-6. doi : 10.1007/978-3-319-46475-6_25. url : https://link.springer.com/chapter/10.1007/978-3-319-46475-6_25 (cf. p. 25).

[42] John Duchi, Elad Hazan et Yoram Singer. « Adaptive Subgradient Methods for Online Learning and Stochastic Optimization ». Dans : *Journal of Machine Learning Research* 12 (Jul 2011), p. 2121-2159. issn : ISSN 1533-7928. url : http://jmlr.org//papers/v12/duchi11a.html (cf. p. 19).

[43] Vincent Dumoulin et Francesco Visin. « A Guide to Convolution Arithmetic for Deep Learning ». Dans : (23 mar. 2016). arXiv : 1603.07285 [cs, stat]. url : http://arxiv.org/abs/1603.07285 (cf. p. 23, 24).

[44] Mark Everingham et al. « The Pascal Visual Object Classes Challenge : A Retrospective ». Dans : *International Journal of Computer Vision* 111.1 (25 juin 2014), p. 98-136. issn : 0920-5691, 1573-1405. doi : 10.1007/s11263-014-0733-5. url : http://link.springer.com/article/10.1007/s11263-014-0733-5 (cf. p. 34-36).

[45] Clément Farabet. « Towards Real-Time Image Understanding with Convolutional Networks ». Université Paris-Est, 2013 (cf. p. 34).

[46] Mathieu Fauvel et al. « Advances in Spectral-Spatial Classification of Hyperspectral Images ». Dans : *Proceedings of the IEEE* 101.3 (mar. 2013), p. 652-675. issn : 0018-9219. doi : 10.1109/JPROC.2012.2197589 (cf. p. 40, 42).

[47] Joseph Fourier. « Propagation de la chaleur dans un solide rectangulaire infini ». Dans : *Théorie analytique de la chaleur*. F. Didot père et fils, 1822, p. 159-177. URL : https://www.bibnum.education.fr/mathematiques/analyse/theorie-analytique-de-la-chaleur (cf. p. 22).

[48] Damien Fourure et al. « Residual Conv-Deconv Grid Network for Semantic Segmentation ». Dans : *BMVC 2017*. Londre, France, sept. 2017. URL : https://hal.archives-ouvertes.fr/hal-01567725 (cf. p. 36).

[49] Bernard Roy Frieden. « A New Restoring Algorithm for the Preferential Enhancement of Edge Gradients ». Dans : *JOSA* 66.3 (1er mar. 1976), p. 280-283. DOI : 10.1364/JOSA.66.000280. URL : https://www.osapublishing.org/josa/abstract.cfm?uri=josa-66-3-280 (cf. p. 22).

[50] Jerome H. Friedman. « Greedy Function Approximation : A Gradient Boosting Machine. » Dans : *The Annals of Statistics* 29.5 (oct. 2001), p. 1189-1232. ISSN : 0090-5364, 2168-8966. DOI : 10.1214/aos/1013203451. URL : https://projecteuclid.org/euclid.aos/1013203451 (cf. p. 41).

[51] Gang Fu et al. « Classification for High Resolution Remote Sensing Imagery Using a Fully Convolutional Network ». Dans : *Remote Sensing* 9.5 (18 mai 2017), p. 498. DOI : 10.3390/rs9050498. URL : http://www.mdpi.com/2072-4292/9/5/498 (cf. p. 43).

[52] Kunihiko Fukushima. « Neocognitron : A Self-Organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position ». Dans : *Biological Cybernetics* 36.4 (1er avr. 1980), p. 193-202. ISSN : 0340-1200, 1432-0770. DOI : 10.1007/BF00344251. URL : https://link.springer.com/article/10.1007/BF00344251 (cf. p. 12, 21).

[53] Ross Girshick et al. « Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Juin 2014, p. 580-587. DOI : 10.1109/CVPR.2014.81 (cf. p. 34).

[54] Ross Girshick et al. « Region-Based Convolutional Networks for Accurate Object Detection and Segmentation ». Dans : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38.1 (jan. 2016), p. 142-158. ISSN : 0162-8828. DOI : 10.1109/TPAMI.2015.2437384 (cf. p. 34).

[55] Xavier Glorot et Yoshua Bengio. « Understanding the Difficulty of Training Deep Feedforward Neural Networks ». Dans : *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics. 31 mar. 2010, p. 249-256. URL : http://proceedings.mlr.press/v9/glorot10a.html (cf. p. 19).

[56] Xavier Glorot, Antoine Bordes et Yoshua Bengio. « Deep Sparse Rectifier Neural Networks ». Dans : *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics. 14 juin 2011, p. 315-323. URL : http://proceedings.mlr.press/v15/glorot11a.html (cf. p. 12, 15).

[57] Pradeep Goel et al. « Classification of Hyperspectral Data by Decision Trees and Artificial Neural Networks to Identify Weed Stress and Nitrogen Status of Corn ». Dans : *Computers and Electronics in Agriculture* 39.2 (mai 2003), p. 67-93. ISSN : 0168-1699. DOI : 10.1016/S0168-1699(03)00020-6. URL : http://www.sciencedirect.com/science/article/pii/S0168169903000206 (cf. p. 41).

[58] Ian Goodfellow, Yoshua Bengio et Aaron Courville. *Deep Learning*. MIT Press, 2016. URL : http://www.deeplearningbook.org (cf. p. 24).

[59] David Grangier, Léon Bottou et Ronan Collobert. « Deep Convolutional Networks for Scene Parsing ». Dans : *ICML 2009 Deep Learning Workshop*. T. 3. Citeseer, 2009 (cf. p. 34).

[60] Chunhui Gu et al. « Recognition Using Regions ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Juin 2009, p. 1030-1037. doi : 10.1109/CVPR.2009.5206727 (cf. p. 34).

[61] Li Guo et al. « Relevance of Airborne Lidar and Multispectral Image Data for Urban Scene Classification Using Random Forests ». Dans : *ISPRS Journal of Photogrammetry and Remote Sensing* 66.1 (1er jan. 2011), p. 56-66. issn : 0924-2716. doi : 10.1016/j.isprsjprs.2010.08.007 (cf. p. 40).

[62] JiSoo Ham et al. « Investigation of the Random Forest Framework for Classification of Hyperspectral Data ». Dans : *IEEE Transactions on Geoscience and Remote Sensing* 43.3 (mar. 2005), p. 492-501. issn : 0196-2892. doi : 10.1109/TGRS.2004.842481 (cf. p. 40).

[63] Kaiming He et al. « Delving Deep into Rectifiers : Surpassing Human-Level Performance on ImageNet Classification ». Dans : *Proceedings of the IEEE International Conference on Computer Vision*. IEEE International Conference on Computer Vision (ICCV). Déc. 2015, p. 1026-1034. doi : 10.1109/ICCV.2015.123 (cf. p. 15, 19).

[64] Kaiming He et al. « Deep Residual Learning for Image Recognition ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, United States, juin 2016, p. 770-778. doi : 10.1109/CVPR.2016.90 (cf. p. 32, 33).

[65] Donald O. Hebb. *The Organization of Behavior*. 1949. pmid : 10643472 (cf. p. 11).

[66] Geoffrey E. Hinton, Simon Osindero et Yee-Whye Teh. « A Fast Learning Algorithm for Deep Belief Nets ». Dans : *Neural Computation* 18.7 (juil. 2006), p. 1527-1554. issn : 0899-7667. doi : 10.1162/neco.2006.18.7.1527. url : http://dx.doi.org/10.1162/neco.2006.18.7.1527 (cf. p. 12, 19).

[67] Geoffrey E. Hinton et Ruslan Salakhutdinov. « Reducing the Dimensionality of Data with Neural Networks ». Dans : *Science* 313.5786 (28 juil. 2006), p. 504-507. issn : 1095-9203. doi : 10.1126/science.1127647. pmid : 16873662 (cf. p. 12).

[68] Sepp Hochreiter et al. « Gradient Flow in Recurrent Nets : The Difficulty of Learning Long-Term Dependencies ». Dans : *Filed Guide to Dynamical Recurrent Networks*. IEEE Press, 2001 (cf. p. 15).

[69] Kurt Hornik. « Approximation Capabilities of Multilayer Feedforward Networks ». Dans : *Neural Networks* 4.2 (1er jan. 1991), p. 251-257. issn : 0893-6080. doi : 10.1016/0893-6080(91)90009-T. url : http://www.sciencedirect.com/science/article/pii/089360809190009T (cf. p. 11, 15).

[70] Andrew G. Howard et al. « MobileNets : Efficient Convolutional Neural Networks for Mobile Vision Applications ». Dans : (16 avr. 2017). arXiv : 1704.04861 [cs]. url : http://arxiv.org/abs/1704.04861 (cf. p. 34).

[71] Bohao Huang et al. « Large-Scale Semantic Classification : Outcome of the First Year of Inria Aerial Image Labeling Benchmark ». Dans : *2018 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. 22 juil. 2018. url : https://hal.inria.fr/hal-01767807/document (cf. p. 43).

[72] Fu Jie Huang et Yann LeCun. « Large-Scale Learning with SVM and Convolutional for Generic Object Categorization ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Juin 2006, p. 284-291. doi : 10.1109/CVPR.2006.164 (cf. p. 12).

[73] Gao Huang et al. « Deep Networks with Stochastic Depth ». Dans : *Computer Vision – ECCV 2016*. Lecture Notes in Computer Science. Springer, Cham, 8 oct. 2016, p. 646-661. ISBN : 978-3-319-46492-3 978-3-319-46493-0. DOI : 10.1007/978-3-319-46493-0_39. URL : https://link.springer.com/chapter/10.1007/978-3-319-46493-0_39 (cf. p. 33).

[74] Gao Huang et al. « Densely Connected Convolutional Networks ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. T. 1. 2017, p. 3 (cf. p. 32, 33).

[75] David H. Hubel et Torsten N. Wiesel. « Receptive Fields of Single Neurones in the Cat's Striate Cortex ». Dans : *The Journal of Physiology* 148.3 (oct. 1959), p. 574-591. ISSN : 0022-3751. pmid : 14403679. URL : https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1363130/ (cf. p. 12).

[76] David H. Hubel et Torsten N. Wiesel. « Receptive Fields and Functional Architecture of Monkey Striate Cortex ». Dans : *The Journal of Physiology* 195.1 (mar. 1968), p. 215-243. ISSN : 0022-3751. pmid : 4966457 (cf. p. 12).

[77] Sergey Ioffe et Christian Szegedy. « Batch Normalization : Accelerating Deep Network Training by Reducing Internal Covariate Shift ». Dans : *Proceedings of the 32nd International Conference on Machine Learning*. International Conference on Machine Learning (ICML). 2015, p. 448-456. URL : http://jmlr.org/proceedings/papers/v37/ioffe15.html (cf. p. 21, 27, 32).

[78] Kevin Jarrett et al. « What Is the Best Multi-Stage Architecture for Object Recognition? » Dans : *Proceedings of the 2009 IEEE 12th International Conference on Computer Vision (ICCV)*. Sept. 2009, p. 2146-2153. DOI : 10.1109/ICCV.2009.5459469 (cf. p. 26, 27).

[79] Simon Jégou et al. « The One Hundred Layers Tiramisu : Fully Convolutional DenseNets for Semantic Segmentation ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Honolulu, United States, juil. 2017, p. 1175-1183. DOI : 10.1109/CVPRW.2017.156 (cf. p. 36).

[80] Judson P. Jones et Larry A. Palmer. « An Evaluation of the Two-Dimensional Gabor Filter Model of Simple Receptive Fields in Cat Striate Cortex ». Dans : *Journal of Neurophysiology* 58.6 (déc. 1987), p. 1233-1258. ISSN : 0022-3077. DOI : 10.1152/jn.1987.58.6.1233. pmid : 3437332 (cf. p. 22).

[81] Diederik Kingma et Jimmy Ba. « Adam : A Method for Stochastic Optimization ». Dans : *Proceedings of the International Conference on Learning Representations (ICLR)*. 2015. arXiv : 1412.6980. URL : http://arxiv.org/abs/1412.6980 (cf. p. 19).

[82] Stephen Cole Kleene. « Representation of Events in Nerve Nets and Finite Automata ». Dans : *Automata Studies*. Princeton University Press (1956), p. 3-42 (cf. p. 10).

[83] Alex Krizhevsky. *Learning Multiple Layers of Features from Tiny Images*. CIFAR, 2009 (cf. p. 28).

[84] Alex Krizhevsky, Ilya Sutskever et Geoffrey E. Hinton. « ImageNet Classification with Deep Convolutional Neural Networks ». Dans : *Proceedings of the Neural Information Processing Systems (NIPS)*. NIPS. 2012, p. 1097-1105. URL : http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf (cf. p. 12, 27, 28, 30).

[85] Anders Krogh et John A. Hertz. « A Simple Weight Decay Can Improve Generalization ». Dans : *Proceedings of the 4th International Conference on Neural Information Processing Systems*. NIPS'91. San Francisco, CA, USA : Morgan Kaufmann Publishers Inc., 1991, p. 950-957. ISBN : 978-1-55860-222-9. URL : http://dl.acm.org/citation.cfm?id=2986916.2987033 (cf. p. 21).

[86] Adrien Lagrange et al. « Benchmarking Classification of Earth-Observation Data : From Learning Explicit Features to Convolutional Networks ». Dans : *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS). Juil. 2015, p. 4173-4176. doi : 10.1109/IGARSS.2015.7326745 (cf. p. 42).

[87] Joseph-Louis Lagrange. *Théorie des fonctions analytiques, contenant les principes du calcul différentiel , dégagés de toute considération d'infiniment petits ou d'évanouissans, de limites ou de fluxions, et réduits a l'analyse algébrique des quantités finies*. À Paris, de l'Imprimerie de la République. Prairial an V., 1797. url : http://gallica.bnf.fr/ark:/12148/bpt6k86263h (cf. p. 17).

[88] David J. Lary et al. « Machine Learning in Geosciences and Remote Sensing ». Dans : *Geoscience Frontiers*. Special Issue : Progress of Machine Learning in Geosciences 7.1 (1er jan. 2016), p. 3-10. issn : 1674-9871. doi : 10.1016/j.gsf.2015.07.003. url : http://www.sciencedirect.com/science/article/pii/S1674987115000821 (cf. p. 41).

[89] Rick Lawrence et al. « Classification of Remotely Sensed Imagery Using Stochastic Gradient Boosting as a Refinement of Classification Tree Analysis ». Dans : *Remote Sensing of Environment* 90.3 (15 avr. 2004), p. 331-336. issn : 0034-4257. doi : 10.1016/j.rse.2004.01.007. url : http://www.sciencedirect.com/science/article/pii/S0034425704000148 (cf. p. 41).

[90] Arthur Le Guennec et al. « Classification de données LiDAR bi-spectral topo-bathymétriques par une approche multi-échelle : Application en milieu fluvial ». Dans : *Conférence Annuelle Française de Photogrammétrie et Télédétection (CFPT)*. Marne-la-Vallée, France, 27 juin 2018, p. 8 (cf. p. 40).

[91] Yann LeCun. « Learning Process in an Asymmetric Threshold Network ». Dans : *Disordered Systems and Biological Organization*. NATO ASI Series. Springer, Berlin, Heidelberg, 1986, p. 233-240. isbn : 978-3-642-82659-7 978-3-642-82657-3. doi : 10.1007/978-3-642-82657-3_24. url : https://link.springer.com/chapter/10.1007/978-3-642-82657-3_24 (cf. p. 11, 16).

[92] Yann LeCun et al. « Backpropagation Applied to Handwritten Zip Code Recognition ». Dans : *Neural Computation* 1.4 (déc. 1989), p. 541-551. issn : 0899-7667. doi : 10.1162/neco.1989.1.4.541 (cf. p. 12).

[93] Yann LeCun et al. « Efficient BackProp ». Dans : *Neural Networks : Tricks of the Trade*. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, 1998, p. 9-50. isbn : 978-3-540-65311-0 978-3-540-49430-0. doi : 10.1007/3-540-49430-8_2. url : https://link.springer.com/chapter/10.1007/3-540-49430-8_2 (cf. p. 15, 17, 20).

[94] Yann LeCun et al. « Gradient-Based Learning Applied to Document Recognition ». Dans : *Proceedings of the IEEE* 86.11 (nov. 1998), p. 2278-2324. issn : 0018-9219. doi : 10.1109/5.726791 (cf. p. 12, 21, 22, 28, 29).

[95] Chen-Yu Lee et al. « Deeply-Supervised Nets ». Dans : *Artificial Intelligence and Statistics*. Artificial Intelligence and Statistics. 21 fév. 2015, p. 562-570. url : http://proceedings.mlr.press/v38/lee15a.html (cf. p. 31, 33).

[96] J. Y. Lettvin et al. « What the Frog's Eye Tells the Frog's Brain ». Dans : *Proceedings of the IRE* 47.11 (nov. 1959), p. 1940-1951. issn : 0096-8390. doi : 10.1109/JRPROC.1959.287207 (cf. p. 12, 13).

[97] Aihua Li et al. « Lidar Aboveground Vegetation Biomass Estimates in Shrublands : Prediction, Uncertainties and Application to Coarser Scales ». Dans : *Remote Sensing* 9.9 (31 août 2017), p. 903. doi : 10.3390/rs9090903. url : http://www.mdpi.com/2072-4292/9/9/903 (cf. p. 40).

[98] Henry W. Lin, Max Tegmark et David Rolnick. « Why Does Deep and Cheap Learning Work So Well? » Dans : *Journal of Statistical Physics* 168.6 (1er sept. 2017), p. 1223-1247. issn : 1572-9613. doi : 10.1007/s10955-017-1836-5. url : https://doi.org/10.1007/s10955-017-1836-5 (cf. p. 16).

[99] Tsung-Yi Lin et al. « Microsoft COCO : Common Objects in Context ». Dans : *Computer Vision – ECCV 2014*. Sous la dir. de David Fleet et al. Lecture Notes in Computer Science 8693. Springer International Publishing, 6 sept. 2014, p. 740-755. isbn : 978-3-319-10601-4 978-3-319-10602-1. doi : 10.1007/978-3-319-10602-1_48. url : http://link.springer.com/chapter/10.1007/978-3-319-10602-1_48 (cf. p. 34, 36).

[100] Cheng-Lin Liu et al. « ICDAR 2011 Chinese Handwriting Recognition Competition ». Dans : *2011 International Conference on Document Analysis and Recognition*. 2011 International Conference on Document Analysis and Recognition. Sept. 2011, p. 1464-1469. doi : 10.1109/ICDAR.2011.291 (cf. p. 12, 28).

[101] Shu Liu et al. « Path Aggregation Network for Instance Segmentation ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Salt Lake City, United States, 2018. arXiv : 1803.01534. url : http://arxiv.org/abs/1803.01534 (cf. p. 36).

[102] Tao Liu et al. « Comparing Fully Convolutional Networks, Random Forest, Support Vector Machine, and Patch-Based Deep Convolutional Neural Networks for Object-Based Wetland Mapping Using Images from Small Unmanned Aircraft System ». Dans : *GIScience & Remote Sensing* 55.2 (4 mar. 2018), p. 243-264. issn : 1548-1603. doi : 10.1080/15481603.2018.1426091. url : https://doi.org/10.1080/15481603.2018.1426091 (cf. p. 43).

[103] Wei Liu et al. « SSD : Single Shot MultiBox Detector ». Dans : *Computer Vision – ECCV 2016*. European Conference on Computer Vision. Lecture Notes in Computer Science. Springer, Cham, 8 oct. 2016, p. 21-37. isbn : 978-3-319-46447-3 978-3-319-46448-0. doi : 10.1007/978-3-319-46448-0_2. url : https://link.springer.com/chapter/10.1007/978-3-319-46448-0_2 (cf. p. 34).

[104] Yansong Liu et al. « Dense Semantic Labeling of Very-High-Resolution Aerial Imagery and LiDAR with Fully-Convolutional Neural Networks and Higher-Order CRFs ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Honolulu, United States, juil. 2017, p. 1561-1570. doi : 10.1109/CVPRW.2017.200 (cf. p. 43).

[105] Jonathan Long, Evan Shelhamer et Trevor Darrell. « Fully Convolutional Networks for Semantic Segmentation ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Juin 2015, p. 3431-3440. doi : 10.1109/CVPR.2015.7298965 (cf. p. 34-36, 43).

[106] Ilya Loshchilov et Frank Hutter. « SGDR : Stochastic Gradient Descent with Warm Restarts ». Dans : *Proceedings of the International Conference on Learning Representations (ICLR)*. 2017 (cf. p. 19).

[107] David G. Lowe. « Object Recognition from Local Scale-Invariant Features ». Dans : *Proceedings of the Seventh IEEE International Conference on Computer Vision*. Proceedings of the Seventh IEEE International Conference on Computer Vision. T. 2. 1999, 1150-1157 vol.2. doi : 10.1109/ICCV.1999.790410 (cf. p. 12, 22, 34).

[108] Zhou Lu et al. « The Expressive Power of Neural Networks : A View from the Width ». Dans : *Advances in Neural Information Processing Systems 30*. Sous la dir. d'I. Guyon et al. Curran Associates, Inc., 2017, p. 6231-6239. url : http://papers.nips.cc/paper/7203-the-expressive-power-of-neural-networks-a-view-from-the-width.pdf (cf. p. 16).

[109] Andrew L. Maas, Awni Y. Hannun et Andrew Y. Ng. « Rectifier Nonlinearities Improve Neural Network Acoustic Models ». Dans : *ICML Workshop on Deep Learning for Audio, Speech and Language Processing*. 2013 (cf. p. 15).

[110] Stéphane Mallat. *Une exploration des signaux en ondelettes*. Palaiseau : Éditions de l'École polytechnique, 12 sept. 2001. 637 p. isbn : 978-2-7302-0733-1 (cf. p. 22).

[111] Stjepan Marčelja. « Mathematical Description of the Responses of Simple Cortical Cells ». Dans : *Journal of the Optical Society of America* 70.11 (1er nov. 1980), p. 1297-1300. doi : 10.1364/JOSA.70.001297. url : https://www.osapublishing.org/josa/abstract.cfm?uri=josa-70-11-1297 (cf. p. 22).

[112] Dimitrios Marmanis et al. « Classification With an Edge : Improving Semantic Image Segmentation with Boundary Detection ». Dans : *ISPRS Journal of Photogrammetry and Remote Sensing* (2017). doi : 10.1016/j.isprsjprs.2017.11.009. arXiv : 1612.01337 (cf. p. 43).

[113] Warren S. McCulloch et Walter H. Pitts. « A Logical Calculus of the Ideas Immanent in Nervous Activity ». Dans : *Bulletin of Mathematical Biophysics* 5 (1943), p. 115-133. url : http://www.cse.chalmers.se/~coquand/AUTOMATA/mcp.pdf (cf. p. 10, 11).

[114] Farid Melgani et Lorenzo Bruzzone. « Classification of Hyperspectral Remote Sensing Images with Support Vector Machines ». Dans : *IEEE Transactions on Geoscience and Remote Sensing* 42.8 (août 2004), p. 1778-1790. issn : 0196-2892. doi : 10.1109/TGRS.2004.831865 (cf. p. 41).

[115] Hrushikesh Mhaskar, Qianli Liao et Tomaso A. Poggio. « When and Why Are Deep Networks Better than Shallow Ones? » Dans : *AAAI*. 2017, p. 2343-2349 (cf. p. 16).

[116] Marvin Minsky et Seymour A. Papert. *Perceptrons*. MIT Press, 1969. url : https://mitpress.mit.edu/books/perceptrons (cf. p. 11).

[117] Volodymyr Mnih. « Machine Learning for Aerial Image Labeling ». University of Toronto, 2013 (cf. p. 42).

[118] Hans Moravec. *Mind Children – The Future of Robot & Human Intelligence*. Cambridge : Harvard University Press, 1988. 224 p. isbn : 978-0-674-57618-6 (cf. p. 28).

[119] Vinod Nair et Geoffrey E. Hinton. « Rectified Linear Units Improve Restricted Boltzmann Machines ». Dans : *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*. 2010, p. 807-814 (cf. p. 15).

[120] Vladimir Nekrasov, Janghoon Ju et Jaesik Choi. « Global Deconvolutional Networks for Semantic Segmentation ». Dans : *British Machine Vision Conference*. 2016. arXiv : 1602.03930. url : http://arxiv.org/abs/1602.03930 (cf. p. 35, 36).

[121] Yurii Nesterov. « A Method of Solving a Convex Programming Problem with Convergence Rate O (1/K2) ». Dans : *Soviet Mathematics Doklady*. T. 27. 1983, p. 372-376 (cf. p. 19).

[122] Gerhard Neuhold et al. « The Mapillary Vistas Dataset for Semantic Understanding of Street Scenes ». Dans : *Proceedings of the International Conference on Computer Vision (ICCV), Venice, Italy*. 2017, p. 22-29 (cf. p. 34).

[123] Keiller Nogueira et al. « Learning to Semantically Segment High-Resolution Remote Sensing Images ». Dans : *2016 23rd International Conference on Pattern Recognition (ICPR)*. 2016 23rd International Conference on Pattern Recognition (ICPR). Déc. 2016, p. 3566-3571. doi : 10.1109/ICPR.2016.7900187 (cf. p. 42).

[124] Hyeonwoo Noh, Seunghoon Hong et Bohyung Han. « Learning Deconvolution Network for Semantic Segmentation ». Dans : *2015 IEEE International Conference on Computer Vision (ICCV)*. 2015 IEEE International Conference on Computer Vision (ICCV). Déc. 2015, p. 1520-1528. doi : 10.1109/ICCV.2015.178 (cf. p. 35).

[125] Avital Oliver et al. « Realistic Evaluation of Semi-Supervised Learning Algorithms ». Dans : *Proceedings of the International Conference on Learning Representations Workshops (ICLR)*. 2018 (cf. p. 20).

[126] Edouard Oyallon. « Building a Regular Decision Boundary with Deep Networks ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, United States, juil. 2017, p. 1886-1894. doi : 10.1109/CVPR.2017.204 (cf. p. 15).

[127] Sakrapee Paisitkriangkrai et al. « Effective Semantic Pixel Labelling with Convolutional Networks and Conditional Random Fields ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Juin 2015, p. 36-43. doi : 10.1109/CVPRW.2015.7301381 (cf. p. 43).

[128] Mahesh Pal. « Random Forest Classifier for Remote Sensing Classification ». Dans : *International Journal of Remote Sensing* 26.1 (1er jan. 2005), p. 217-222. issn : 0143-1161. doi : 10.1080/01431160412331269698. url : https://doi.org/10.1080/01431160412331269698 (cf. p. 41).

[129] Mahesh Pal et Paul M. Mather. « Support Vector Machines for Classification in Remote Sensing ». Dans : *International Journal of Remote Sensing* 26.5 (1er mar. 2005), p. 1007-1011. issn : 0143-1161. doi : 10.1080/01431160512331314083. url : https://doi.org/10.1080/01431160512331314083 (cf. p. 41).

[130] Maria Papadomanolaki, Maria Vakalopoulou et Konstantinos Karantzalos. « Patch-Based Deep Learning Architectures for Sparse Annotated Very High Resolution Datasets ». Dans : *2017 Joint Urban Remote Sensing Event (JURSE)*. 2017 Joint Urban Remote Sensing Event (JURSE). Mar. 2017, p. 1-4. doi : 10.1109/JURSE.2017.7924538 (cf. p. 42).

[131] Constantine P. Papageorgiou, Michael Oren et Tomaso Poggio. « A General Framework for Object Detection ». Dans : *Proceedings of the Sixth International Conference on Computer Vision*. ICCV '98. Washington, DC, USA : IEEE Computer Society, 1998, p. 555-. isbn : 978-81-7319-221-0. url : http://dl.acm.org/citation.cfm?id=938978.939174 (cf. p. 22).

[132] Seymour Papert. *The Summer Vision Project*. 1966 (cf. p. 10, 28).

[133] Peeta Basa Pati et A. G. Ramakrishnan. « Word Level Multi-Script Identification ». Dans : *Pattern Recognition Letters* 29.9 (1er juil. 2008), p. 1218-1229. issn : 0167-8655. doi : 10.1016/j.patrec.2008.01.027. url : http://www.sciencedirect.com/science/article/pii/S0167865508000354 (cf. p. 22).

[134] Chao Peng et al. « Large Kernel Matters – Improve Semantic Segmentation by Global Convolutional Network ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Juil. 2017, p. 4353-4361. url : http://openaccess.thecvf.com/content_cvpr_2017/html/Peng_Large_Kernel_Matters_CVPR_2017_paper.html (cf. p. 36).

[135] M. T. Pham, E. Aptoula et S. Lefèvre. « Feature Profiles from Attribute Filtering for Classification of Remote Sensing Images ». Dans : *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 11.1 (jan. 2018), p. 249-256. issn : 1939-1404. doi : 10.1109/JSTARS.2017.2773367 (cf. p. 43).

[136] Nicolas Pinto, David D. Cox et James J. DiCarlo. « Why Is Real-World Visual Object Recognition Hard? » Dans : *PLOS Computational Biology* 4.1 (25 jan. 2008), e27. issn : 1553-7358. doi : 10.1371/journal.pcbi.0040027. url : http://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.0040027 (cf. p. 26).

[137] Tomaso Poggio et al. « Why and When Can Deep-but Not Shallow-Networks Avoid the Curse of Dimensionality : A Review ». Dans : *International Journal of Automation and Computing* 14.5 (1er oct. 2017), p. 503-519. ISSN : 1476-8186, 1751-8520. DOI : 10.1007/s11633-017-1054-2. URL : https://link.springer.com/article/10.1007/s11633-017-1054-2 (cf. p. 16).

[138] Boris Polyak et Anatoli Juditsky. « Acceleration of Stochastic Approximation by Averaging ». Dans : *SIAM Journal on Control and Optimization* 30.4 (juil. 1992), p. 838-855. ISSN : 0363-0129. DOI : 10.1137/0330046. URL : http://dx.doi.org/10.1137/0330046 (cf. p. 19).

[139] Ning Qian. « On the Momentum Term in Gradient Descent Learning Algorithms ». Dans : *Neural Networks* 12.1 (1er jan. 1999), p. 145-151. ISSN : 0893-6080. DOI : 10.1016/S0893-6080(98)00116-6. URL : https://www.sciencedirect.com/science/article/pii/S0893608098001166 (cf. p. 19).

[140] Rajat Raina, Anand Madhavan et Andrew Y. Ng. « Large-Scale Deep Unsupervised Learning Using Graphics Processors ». Dans : *Proceedings of the 26th Annual International Conference on Machine Learning*. ICML '09. New York, NY, USA : ACM, 2009, p. 873-880. ISBN : 978-1-60558-516-1. DOI : 10.1145/1553374.1553486. URL : http://doi.acm.org/10.1145/1553374.1553486 (cf. p. 12).

[141] Olaf Ronneberger, Philipp Fischer et Thomas Brox. « U-Net : Convolutional Networks for Biomedical Image Segmentation ». Dans : *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015 : 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III*. Sous la dir. de Nassir Navab et al. Cham : Springer International Publishing, 2015, p. 234-241. ISBN : 978-3-319-24574-4. DOI : 10.1007/978-3-319-24574-4_28. URL : https://doi.org/10.1007/978-3-319-24574-4_28 (cf. p. 35, 36).

[142] Frank Rosenblatt. *The Perceptron : A Probabilistic Model for Information Storage and Organization In The Brain*. 1957 (cf. p. 11, 28).

[143] J. W. Rouse. « Monitoring Vegetation Systems in the Great Plains with ERTS ». Dans : 1er jan. 1974. URL : https://ntrs.nasa.gov/search.jsp?R=19740022614 (cf. p. 40).

[144] D. E. Rumelhart, G. E. Hinton et R. J. Williams. « Learning Internal Representations by Error Propagation ». Dans : sous la dir. de David E. Rumelhart, James L. McClelland et CORPORATE PDP Research Group. Cambridge, MA, USA : MIT Press, 1986, p. 318-362. ISBN : 978-0-262-68053-0. URL : http://dl.acm.org/citation.cfm?id=104279.104293 (cf. p. 11, 16, 17).

[145] Olga Russakovsky et al. « ImageNet Large Scale Visual Recognition Challenge ». Dans : *International Journal of Computer Vision* 115.3 (11 avr. 2015), p. 211-252. ISSN : 0920-5691, 1573-1405. DOI : 10.1007/s11263-015-0816-y. URL : http://link.springer.com/article/10.1007/s11263-015-0816-y (cf. p. 28, 30).

[146] Ruslan Salakhutdinov et Geoffrey Hinton. « Deep Boltzmann Machines ». Dans : *Artificial Intelligence and Statistics*. Artificial Intelligence and Statistics. 15 avr. 2009, p. 448-455. URL : http://proceedings.mlr.press/v5/salakhutdinov09a.html (cf. p. 12).

[147] Shibani Santurkar et al. « How Does Batch Normalization Help Optimization ? (No, It Is Not About Internal Covariate Shift) ». Dans : (29 mai 2018). arXiv : 1805.11604 [cs, stat]. URL : http://arxiv.org/abs/1805.11604 (cf. p. 27).

[148] Andrew M. Saxe, James L. McClelland et Surya Ganguli. « Exact Solutions to the Nonlinear Dynamics of Learning in Deep Linear Neural Networks ». Dans : *Proceedings of the International Conference on Learning Representations (ICLR)*. 2014. arXiv : 1312.6120 (cf. p. 18, 20).

[149] Henry Schneiderman et Takeo Kanade. « Probabilistic Modeling of Local Appearance and Spatial Relationships for Object Recognition ». Dans : *Computer Vision and Pattern Recognition, 1998. Proceedings. 1998 IEEE Computer Society Conference On*. IEEE, 1998, p. 45-51 (cf. p. 28).

[150] Pierre Sermanet et al. « OverFeat : Integrated Recognition, Localization and Detection Using Convolutional Networks ». Dans : *Proceedings of the International Conference on Learning Representations (ICLR)*. 2014. arXiv : 1312.6229. url : http://arxiv.org/abs/1312.6229 (cf. p. 34).

[151] T. Serre, L. Wolf et T. Poggio. « Object Recognition with Features Inspired by Visual Cortex ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. T. 2. Juin 2005, 994-1000 vol. 2. doi : 10.1109/CVPR.2005.254 (cf. p. 12).

[152] Thomas Serre et al. « A Quantitative Theory of Immediate Visual Recognition ». Dans : *Progress in Brain Research* 165 (2007), p. 33-56. issn : 0079-6123. doi : 10.1016/S0079-6123(06)65004-8. pmid : 17925239 (cf. p. 12).

[153] M. J. Shensa. « The Discrete Wavelet Transform : Wedding the a Trous and Mallat Algorithms ». Dans : *IEEE Transactions on Signal Processing* 40.10 (oct. 1992), p. 2464-2482. issn : 1053-587X. doi : 10.1109/78.157290 (cf. p. 25).

[154] Jamie Sherrah. « Fully Convolutional Networks for Dense Semantic Labelling of High-Resolution Aerial Imagery ». Dans : (8 juin 2016). arXiv : 1606.02585 [cs]. url : http://arxiv.org/abs/1606.02585 (cf. p. 43).

[155] Jamie Shotton, Matthew Johnson et Roberto Cipolla. « Semantic Texton Forests for Image Categorization and Segmentation ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2008, p. 1-8. doi : 10.1109/CVPR.2008.4587503 (cf. p. 34).

[156] Jamie Shotton et al. « Real-Time Human Pose Recognition in Parts from Single Depth Images ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2011, p. 1297-1304. doi : 10.1109/CVPR.2011.5995316 (cf. p. 34).

[157] Karen Simonyan et Andrew Zisserman. « Very Deep Convolutional Networks for Large-Scale Image Recognition ». Dans : *Proceedings of the International Conference on Learning Representations (ICLR)*. Mai 2015. url : http://arxiv.org/abs/1409.1556 (cf. p. 31, 32, 35).

[158] Irwin Sobel. « An Isotropic 3x3 Image Gradient Operator ». Dans : *Presentation at Stanford A.I. Project 1968* (8 fév. 2014) (cf. p. 22).

[159] Sho Sonoda et Noboru Murata. « Neural Network with Unbounded Activation Functions Is Universal Approximator ». Dans : *Applied and Computational Harmonic Analysis* 43.2 (1er sept. 2017), p. 233-268. issn : 1063-5203. doi : 10.1016/j.acha.2015.12.005. url : http://www.sciencedirect.com/science/article/pii/S1063520315001748 (cf. p. 15).

[160] Nitish Srivastava et al. « Dropout : A Simple Way to Prevent Neural Networks from Overfitting ». Dans : *Journal of Machine Learning Research* 15 (2014), p. 1929-1958. url : http://jmlr.org/papers/v15/srivastava14a.html (cf. p. 21, 28, 32).

[161] J. Stallkamp et al. « The German Traffic Sign Recognition Benchmark : A Multi-Class Classification Competition ». Dans : *The 2011 International Joint Conference on Neural Networks*. The 2011 International Joint Conference on Neural Networks. Juil. 2011, p. 1453-1460. doi : 10.1109/IJCNN.2011.6033395 (cf. p. 12, 28).

[162] Ilya Sutskever et al. « On the Importance of Initialization and Momentum in Deep Learning ». Dans : Proceedings of The 30th International Conference on Machine Learning. 2013, p. 1139-1147. URL : http://jmlr.org/proceedings/papers/v28/sutskever13.html (cf. p. 19).

[163] Christian Szegedy et al. « Going Deeper with Convolutions ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Juin 2015, p. 1-9. DOI : 10.1109/CVPR.2015.7298594. URL : http://www.cv-foundation.org/openaccess/content_cvpr_2015/html/Szegedy_Going_Deeper_With_2015_CVPR_paper.html (cf. p. 31, 32, 35, 36).

[164] Christian Szegedy et al. « Rethinking the Inception Architecture for Computer Vision ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Juin 2016, p. 2818-2826. DOI : 10.1109/CVPR.2016.308 (cf. p. 31).

[165] Christian Szegedy et al. « Inception-v4, Inception-Resnet and the Impact of Residual Connections on Learning. » Dans : *AAAI Conference on Artificial Intelligence*. AAAI Conference on Artificial Intelligence. T. 4. 2017, p. 12 (cf. p. 33).

[166] Richard Szeliski. *Computer Vision : Algorithms and Applications*. Texts in Computer Science. London : Springer-Verlag, 2011. ISBN : 978-1-84882-934-3. URL : //www.springer.com/us/book/9781848829343 (cf. p. 28).

[167] Tijmen Tielman et Geoffrey Hinton. *Lecture 6.5—RmsProp : Divide the Gradient by a Running Average of Its Recent Magnitude*. 2012 (cf. p. 19).

[168] A. M. Turing. *Computing Machinery and Intelligence*. 1950 (cf. p. 10).

[169] J. R. R. Uijlings et al. « Selective Search for Object Recognition ». Dans : *International Journal of Computer Vision* 104.2 (1er sept. 2013), p. 154-171. ISSN : 0920-5691, 1573-1405. DOI : 10.1007/s11263-013-0620-5. URL : https://link.springer.com/article/10.1007/s11263-013-0620-5 (cf. p. 34).

[170] Shimon Ullman. « Aligning Pictorial Descriptions : An Approach to Object Recognition ». Dans : *Cognition* 32.3 (1er août 1989), p. 193-254. ISSN : 0010-0277. DOI : 10.1016/0010-0277(89)90036-X. URL : http://www.sciencedirect.com/science/article/pii/001002778990036X (cf. p. 28).

[171] Dmitry Ulyanov, Andrea Vedaldi et Victor Lempitsky. « Deep Image Prior ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Salt Lake City, United States, juin 2018. arXiv : 1711.10925. URL : http://arxiv.org/abs/1711.10925 (cf. p. 24).

[172] M. Vakalopoulou et al. « Building Detection in Very High Resolution Multispectral Data with Deep Learning Features ». Dans : *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS). Juil. 2015, p. 1873-1876. DOI : 10.1109/IGARSS.2015.7326158 (cf. p. 42).

[173] John E. Vargas et al. « Superpixel-Based Interactive Classification of Very High Resolution Images ». Dans : *27th SIBGRAPI Conference on Graphics, Patterns and Images*. Août 2014, p. 173-179. DOI : 10.1109/SIBGRAPI.2014.49 (cf. p. 42).

[174] Andreas Veit, Michael Wilber et Serge Belongie. « Residual Networks Behave Like Ensembles of Relatively Shallow Networks ». Dans : *Proceedings of the 30th International Conference on Neural Information Processing Systems*. NIPS'16. USA : Curran Associates Inc., 2016, p. 550-558. ISBN : 978-1-5108-3881-9. URL : http://dl.acm.org/citation.cfm?id=3157096.3157158 (cf. p. 33).

[175] Michel Vidal-Naquet et Shimon Ullman. « Object Recognition with Informative Features and Linear Classification. » Dans : *ICCV*. T. 3. 2003, p. 281 (cf. p. 28).

[176] Paul Viola et Michael Jones. « Robust Real-Time Object Detection ». Dans : *International Journal of Computer Vision*. 2001 (cf. p. 22, 34).

[177] Michele Volpi et Devis Tuia. « Dense Semantic Labeling of Subdecimeter Resolution Images With Convolutional Neural Networks ». Dans : *IEEE Transactions on Geoscience and Remote Sensing* 55.2 (fév. 2017), p. 881-893. ISSN : 0196-2892. DOI : 10.1109/TGRS.2016.2616585 (cf. p. 43).

[178] Li Wan et al. « Regularization of Neural Networks Using DropConnect ». Dans : *International Conference on Machine Learning*. International Conference on Machine Learning. 13 fév. 2013, p. 1058-1066. URL : http://proceedings.mlr.press/v28/wan13.html (cf. p. 21).

[179] Paul John Werbos. « Beyond Regression : New Tools for Prediction and Analysis in the Behavioral Sciences ». Harvard University, 1975. 906 p. (cf. p. 11, 16, 17).

[180] Bernard Widrow. *An Adaptive "ADALINE" Neuron Using Chemical "Memistors"*. Stanford University, 17 oct. 1960. URL : http://www-isl.stanford.edu/~widrow/papers/t1960anadaptive.pdf (cf. p. 11).

[181] Rodney Winter et Bernard Widrow. « MADALINE RULE II : A Training Algorithm for Neural Networks ». Dans : *IEEE 1988 International Conference on Neural Networks*. IEEE 1988 International Conference on Neural Networks. Juil. 1988, 401-408 vol.1. DOI : 10.1109/ICNN.1988.23872 (cf. p. 11).

[182] Zifeng Wu, Chunhua Shen et Anton Van Den Hengel. « High-Performance Semantic Segmentation Using Very Deep Fully Convolutional Networks ». Dans : (14 avr. 2016). arXiv : 1604.04339 [cs]. URL : http://arxiv.org/abs/1604.04339 (cf. p. 36).

[183] Huan Xie et al. « New Hyperspectral Difference Water Index for the Extraction of Urban Water Bodies by the Use of Airborne Hyperspectral Images ». Dans : *Journal of Applied Remote Sensing* 8.1 (juil. 2014), p. 085098. ISSN : 1931-3195, 1931-3195. DOI : 10.1117/1.JRS.8.085098. URL : https://www.spiedigitallibrary.org/journals/Journal-of-Applied-Remote-Sensing/volume-8/issue-1/085098/New-hyperspectral-difference-water-index-for-the-extraction-of-urban/10.1117/1.JRS.8.085098.short (cf. p. 40).

[184] Jason Yosinski et al. « How Transferable Are Features in Deep Neural Networks ? » Dans : *Advances in Neural Information Processing Systems*. Neural Information Processing Systems (NIPS). 2014, p. 3320-3328. URL : http://papers.nips.cc/paper/5347-how-transferable-are-features-in-deep-neural-networks (cf. p. 22).

[185] Fisher Yu et Vladlen Koltun. « Multi-Scale Context Aggregation by Dilated Convolutions ». Dans : *Proceedings of the International Conference on Learning Representations (ICLR)*. 23 nov. 2015. URL : http://arxiv.org/abs/1511.07122 (cf. p. 25, 35, 36).

[186] Matthew D. Zeiler. « ADADELTA : An Adaptive Learning Rate Method ». Dans : (22 déc. 2012). arXiv : 1212.5701 [cs]. URL : http://arxiv.org/abs/1212.5701 (cf. p. 19).

[187] Matthew Zeiler et Rob Fergus. « Stochastic Pooling for Regularization of Deep Convolutional Neural Networks ». Dans : *Proceedings of the International Conference on Learning Representations (ICLR)*. 2013. URL : https://openreview.net/forum?id=1_PClqDdLb5Bp (cf. p. 21).

[188] Matthew Zeiler et Rob Fergus. « Visualizing and Understanding Convolutional Networks ». Dans : *Computer Vision–ECCV 2014*. Springer, 2014, p. 818-833. URL : http://link.springer.com/chapter/10.1007/978-3-319-10590-1_53 (cf. p. 30).

[189] Hengshuang Zhao et al. « Pyramid Scene Parsing Network ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, United States, juil. 2017, p. 2881-2890. DOI : 10.1109/CVPR.2017.660. URL : http://openaccess.thecvf.com/content_cvpr_2017/html/Zhao_Pyramid_Scene_Parsing_CVPR_2017_paper.html (cf. p. 36).

[190] Junbo Zhao et al. « Stacked What-Where Auto-Encoders ». Dans : *Proceedings of the International Conference on Learning Representations (ICLR)*. 8 juin 2015. URL : http://arxiv.org/abs/1506.02351 (cf. p. 25, 35).

[191] Shuai Zheng et al. « Conditional Random Fields as Recurrent Neural Networks ». Dans : *Proceedings of the IEEE International Conference on Computer Vision*. IEEE International Conference on Computer Vision (ICCV). Déc. 2015, p. 1529-1537. DOI : 10.1109/ICCV.2015.179 (cf. p. 36).

[192] Bolei Zhou et al. « Scene Parsing through ADE20K Dataset ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, United States, juil. 2017, p. 5122-5130. DOI : 10.1109/CVPR.2017.544 (cf. p. 36).

[193] Yi-Tong Zhou et Rama Chellappa. « Stereo Matching Using a Neural Network ». Dans : *ICASSP-88., International Conference on Acoustics, Speech, and Signal Processing*. ICASSP-88., International Conference on Acoustics, Speech, and Signal Processing. Avr. 1988, 940-943 vol.2. DOI : 10.1109/ICASSP.1988.196745 (cf. p. 26).

[194] Barret Zoph et Quoc Le. « Neural Architecture Search with Reinforcement Learning ». Dans : *Proceedings of the International Conference on Learning Representations (ICLR)*. 2017 (cf. p. 16).

[195] Will Zou et al. « Generic Object Detection with Dense Neural Patterns and Regionlets ». Dans : *Proceedings of the British Machine Vision Conference*. BMVA Press, 2014, p. 72.1-72.11. ISBN : 978-1-901725-52-0. DOI : 10.5244/C.28.72. URL : http://www.bmva.org/bmvc/2014/papers/paper050/index.html (cf. p. 34).

Chapitre 3

Cartographie automatisée d'images aériennes

Et la géographie, c'est exact, m'a beaucoup servi. Je savais reconnaître, du premier coup d'œil, la Chine de l'Arizona. C'est très utile, si l'on est égaré pendant la nuit.

— Antoine de Saint-Exupéry (Le Petit Prince, 1943)

Sommaire

3.1 Classification par région d'images aériennes 60
3.1.1 Classification par région 61
3.1.2 Algorithmes de segmentation 61
3.1.3 Choix de la méthode de segmentation 63
3.2 Réseaux de neurones profonds 66
3.2.1 Réseaux de neurones convolutifs comme extracteurs de caractéristiques 66
3.2.2 Réseaux de neurones entièrement convolutifs 67
3.2.3 Aspects multiéchelles 69
3.3 Évaluation des modèles 72
3.3.1 Métriques pour la classification 72
3.3.2 Métriques pour la segmentation 73
3.3.3 Classification par région 74
3.3.4 Classification pixellique par segmentation sémantique 76

Résumé du chapitre :

Ce chapitre présente deux approches de segmentation sémantique d'images aériennes à très haute résolution : la classification par région et les réseaux entièrement convolutifs.

La classification par région consiste à partitionner l'image en sous-parties homogènes via un algorithme de segmentation non-supervisé. Les régions ainsi obtenues sont ensuite classifiées une à une. Pour ce faire, nous extrayons des caractéristiques profondes sur chaque sous-image à l'aide de CNN préentraînés sur ImageNet et montrons que ces représentations apprises à partir d'images multimédia peuvent se transférer avec succès pour l'analyse d'images aériennes.

En outre, nous identifions les propriétés souhaitables des segmentations non-supervisées impliquées dans ce processus de classification par région. Nous mettons en évidence le rôle limitant de la sous-segmentation, aussi bien pour l'extraction de caractéristiques que pour la segmentation, ne pouvant être compensé que par une coûteuse diminution de la taille des régions. Nous proposons alors d'introduire les réseaux de neurones entièrement convolutifs pour la télédétection, capables de réaliser une extraction de caractéristiques et une classification dense sur tous les pixels d'une image en une seule inférence.

Nous adaptons plusieurs modèles de l'état de l'art pour la segmentation sémantique d'images naturelles aux données de télédétection, sur lesquelles nous montrons la supériorité empirique des réseaux entièrement convolutifs par rapport aux méthodes de classification usuelles. Nous étudions enfin plusieurs variantes multiéchelles permettant de prendre en compte différents niveaux de contexte spatial.

3.1 Classification par région d'images aériennes

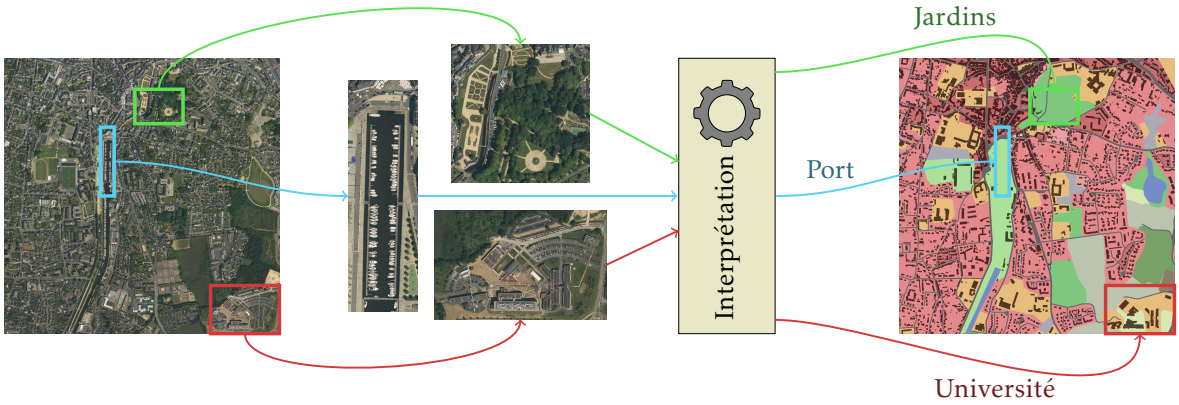


FIGURE 3.1 – Cartographie automatisée d'images aériennes.

Ce chapitre s'intéresse à la cartographie d'images aériennes à trois canaux, rouge-vert-bleu (RVB) ou infra-rouge-rouge-vert (IRRV), en THR ($< 50cm$) ou EHR ($< 10cm$). En effet, celles-ci présentent des caractéristiques techniques proches des images multimédia habituellement manipulées en vision par ordinateur : résolution élevée, espace de couleur RVB ou assimilé et acquises par des appareils photo traditionnels. Il s'agit donc d'une première étape naturelle pour l'interprétation d'images de télédétection.

Nous souhaitons apprendre un modèle de cartographie sémantique à partir des images, c'est-à-dire l'association de chaque élément de l'image considérée à une classe d'intérêt (Figure 3.1). Formellement, il s'agit pour une image I de dimensions $M \times N$ et un ensemble d'étiquettes 1 à n d'associer à chaque pixel $I_{i,j}$ une classe $k_{i,j} \in \{1,\dots,n\}$. Nous allons donc chercher à approcher la fonction f telle que :

$$\forall (i,j) \in \{1 \dots M\} \times \{1 \dots N\} \quad f(I[i,j]) = k_{i,j} \,. \tag{3.1}$$

Contrairement au problème de la reconnaissance d'objet, qui associe une ou plusieurs étiquettes à une image dans son intégralité, il s'agit ici d'un problème de classification *dense*. En raison des relations spatiales existant entre les pixels $I[i,j]$, l'image ainsi classifiée se représente sous la forme de groupes de pixels voisins appartenant à la même classe. Ce problème se trouve généralement dans la littérature sous la dénomination « segmentation sémantique ».

Afin de construire un modèle statistique approché de f, il est possible de la décomposer en deux fonctions successives. La première étape, dite d'extraction de caractéristiques, consiste en une projection de l'information initiale dans espace de représentation choisi au préalable. La seconde consiste à diviser l'espace ainsi formé en sous-ensemble disjoints, c'est-à-dire à réaliser la classification à proprement parler.

Formellement, en notant $\mathcal{E}$ l'espace d'entrée du modèle, $\mathcal{R}$ l'espace de représentation et $\{y_1,\dots,y_k\}$ les classes d'intérêt, nous décomposons donc f sous la forme suivante :

$$f = c \circ p \tag{3.2}$$

avec p une projection de $\mathcal{E} \to \mathcal{R}$ et $c : \mathcal{R} \to \{y_1, \dots y_k\}$ de telle sorte que :

$$\forall x \in \mathcal{E} \text{ une combinaison de pixels}, f(x) = c(p(x)) = y \in \{y_1,\dots,y_k\} \,. \tag{3.3}$$

Le choix de l'espace de représentation, et donc de la projection p, est rarement entièrement décorrélé de celui du classifieur. Par exemple, une SVM à noyau linéaire partitionne

l'espace de représentation en déterminant les hyperplans séparateurs maximisant la distance aux données. Idéalement, on cherchera donc à ce que l'image de l'espace d'entrée $\mathcal{E}$ par p soit linéairement séparable dans $\mathcal{R}$. Par la suite, nous appellerons les éléments de $\mathcal{R}$ des caractéristiques et p sera identifié comme extracteur de caractéristiques.

La suite de cette section introduit le problème de classification par région avant de rappeler l'état de l'art en segmentation non supervisée et d'étudier les propriétés de tels algorithmes lorsqu'ils sont appliqués sur des images aériennes.

3.1.1 Classification par région

Comme nous l'avons vu dans le chapitre précédent, la classification d'images est un domaine ayant largement été exploré dans la littérature. Toutefois, notre intérêt ici se porte non pas sur l'association d'une image à une classe, mais à la mise en correspondance de chaque pixel de l'image avec une étiquette sémantique. Une première façon d'aborder cette tâche consiste à séparer la segmentation d'une part et la sémantisation d'autre part. Ce mécanisme permet de découper l'image en plusieurs morceaux qui seront ensuite classifiés séparément, on parle alors de classification par région. Dans un premier temps, un algorithme de segmentation partitionne l'image, puis un classifieur assigne une classe à chacune des sous-parties identifiées.

Définition 6. *La classification par région d'une image* I *consiste à trouver une partition* $\mathrm{P} = \mathrm{P}_1, \ldots, \mathrm{P}_n$ *telle que :*

$$\bigcup_{i=1}^{n} \mathrm{P}_i = \mathrm{I} \quad \textit{(segmentation)}$$

et une fonction de classification $\mathcal{C}$ *telle que :*

$$\mathcal{C}(\mathrm{P}_i) = k_i \quad \textit{(classification)}$$

avec k_i *la classe d'intérêt associée à la* i^e *région* [1].

Diverses approches ont été proposées dans la littérature en télédétection, utilisant par exemple des profils d'attributs sur des segmentations hiérarchiques arborescentes [9], des segmentations de type superpixels combinées à une approche par sac de mots visuels [34] ou encore des réseaux de neurones profonds [23]. En vision par ordinateur, Couprie et al. [18] exploitaient une segmentation non-supervisée afin de régulariser la segmentation sémantique d'images Red-Green-Blue + Depth (RGB-D). Dans un premier temps, nous passons en revue les méthodes de segmentation non-supervisée et nous étudions l'influence de celles-ci sur les performances des classifieurs utilisés dans le cadre de la classification par région.

3.1.2 Algorithmes de segmentation

Il existe de nombreux algorithmes de segmentation d'image non-supervisés dont le champ d'application varie des images monochromes en niveaux de gris jusqu'aux représentations colorées dans les espaces RVB, teinte-saturation-intensité ou encore L*a*b* CIE 1976 (CIELAB).

Une première famille d'algorithmes de segmentation considère l'image comme un graphe. Formellement, les pixels sont représentés par les nœuds du graphe dont les arêtes représentent les relations de similarité entre voisins. La construction des régions de l'image se fait alors en agglomérant les nœuds du graphe en fonction des arêtes qui les relient. C'est sur ce principe que fonctionne l'algorithme de segmentation Felzenszwalb-Huttenlocher (FH) [20], qui segmente l'image en calculant un arbre couvrant de poids minimal, mais aussi l'algorithme *Normalized Cuts* [57] qui aborde le problème sous l'angle du partitionnement de

1. Il s'agit généralement de la classe majoritairement représentée dans la région.



FIGURE 3.2 – Segmentations d'une image naturelle. Certains algorithmes produisent des régions irrégulières, mais capturant mieux les détails de l'image. Crédits image : Tom Frydenlund, CC0.

graphe. C'est également l'approche utilisée pour les algorithmes de marche aléatoire pour la minimisation d'une fonction entropie *Entropy Rate Superpixel* (ERS) [39] et pour la résolution d'une équation de diffusion [25].

Une seconde approche, à la popularité croissante, dérive des algorithmes itératifs de *clustering* (partitionnement de données). Ce procédé a engendré deux grandes familles de segmentations dites « superpixels ». La première est dérivée de l'algorithme *Simple Linear Iterative Clustering* (SLIC) [1]. Cet algorithme projette les pixels dans un espace de représentation couleur-(x, y) de dimension 5 et utilise un algorithme de partitionnement itératif dérivé des k-moyennes. SLIC initialise un nombre de centres déterminé par l'utilisateur sur une grille régulière, puis met à jour itérativement ceux-ci en absorbant les pixels voisins de la frontière des régions segmentées. Cette méthode a vu naître plusieurs variantes, dont le *Preemptive SLIC* [48], plus rapide, l'algorithme *Linear Spectral Clustering* (LSC) [35] intégrant des contraintes globales en plus de la mise à jour itérative locale et l'algorithme *Superpixels with Contour Adherence using Linear Path* (SCALP) [22]. SCALP interdit l'apparition de superpixels de forme non-régulière dans l'image en considérant l'ensemble des pixels sur le chemin entre le barycentre du superpixel et celui à ajouter. En outre, SCALP prend en entrée le résultat d'un algorithme de détection de contours afin de renforcer l'adhérence des super-

pixels aux bordures des objets. La deuxième grande famille d'algorithmes de segmentation en superpixels se base sur le principe des k-médoïdes. En particulier, il s'agit de projeter les pixels dans un espace non-euclidien de dimension 5 (généralement RVB-(x,y)) puis de réaliser le partitionnement en cherchant le mode dominant local de chaque voisinage, c'est-à-dire la médoïde. Cette approche a notamment été utilisée pour les algorithmes *Mean Shift* [17] et *Quickshift* [64]. Plusieurs autres algorithmes utilisent également des approches itératives de partitionnement. L'algorithme *Superpixels Extracted via Energy-Driven Sampling* (SEEDS) [62] définit ainsi des blocs de pixels capables d'échanger des éléments le long de leur frontière afin de maximiser une fonction d'énergie dépendant des histogrammes de couleurs. SEEDS utilise une optimisation par *hill-climbing* afin de faire converger itérativement les blocs vers une segmentation stable. Enfin, il existe également des algorithmes itératifs convergeant vers une segmentation à partir de la méthode des surfaces de niveau, comme l'algorithme de Chan-Vese [14] dérivé des contours actifs ou l'algorithme *TurboPixel* [33] considérant localement la courbure et le gradient de l'image.

Pour la segmentation d'images en niveaux de gris, l'approche morphologique *watershed* (ou segmentation par ligne de partage des eaux) [8] est particulièrement populaire. *Watershed* considère l'image comme une carte d'élévation dans laquelle est simulée la montée du niveau de l'eau. Initialement, l'eau s'écoule depuis un certain de nombre de sources positionnées sur des marqueurs, qui peuvent être insérés manuellement ou calculés automatiquement [2]. L'eau remplit alors le relief topographique et le niveau est artificiellement augmenté. Lorsque deux sources se rencontrent, un barrage virtuel est érigé à leur ligne de démarcation, établissant ainsi une des frontières de la segmentation. L'algorithme s'arrête lorsque toute l'image a été inondée. Le choix des marqueurs initiaux de *watershed* est crucial pour la qualité de la segmentation et notamment pour la régularité des régions produites. Une version dite compacte a été proposée [48] afin de rendre *watershed* robuste à l'initialisation, en la rendant plus proche de SLIC. L'approche morphologique peut également être utilisée dans le cadre des contours actifs, notamment dans une variante de l'algorithme Chan-Vese [14] utilisant les contours actifs morphologiques [46].

Enfin, des algorithmes spécifiques au traitement d'images de télédétection, notamment radar et multispectrales, ont été proposés dans la littérature. Ces segmentations prennent notamment en compte des aspects multiéchelles avec pour objectif final l'analyse d'image orientée objet. Ainsi, l'algorithme *Multi-Resolution Segmentation* (MRS) [5] est une méthode populaire de segmentation d'images de télédétection, notamment grâce à son implémentation dans le logiciel eCognition©. MRS se focalise sur l'identification d'objets saillants et utilise une approche par croissance de régions selon un critère d'homogénéité spectrale défini de façon heuristique. La segmentation est exécutée à plusieurs échelles, un critère de similarité *ad hoc* déterminant la conservation ou la fusion des régions les plus fines. L'algorithme *Hierarchial Segmentation* (HSeg) [61] produit quant à lui une segmentation hiérarchique multiéchelle arborescente : une région de l'échelle la plus grande est sous-divisée en plusieurs régions, elles-mêmes pouvant être divisées récursivement. HSeg utilise une approche par croissance de régions dans laquelle les pixels proches sont itérativement fusionnés à moins de vérifier un critère spécifique de dissimilarité. Des régions voisines peuvent ensuite être elles-même fusionnées en cas d'homogénéité, afin de produire une segmentation hiérarchique à une échelle plus faible.

3.1.3 Choix de la méthode de segmentation

La profusion de méthodes de segmentation existantes pose la question du choix de celle la plus adaptée pour l'apprentissage statistique. Deux critères sont à prendre en compte : quels prétraitements est-il nécessaire d'appliquer à l'image, et quelle segmentation semble respecter au mieux les propriétés spatiales de l'image ?

2. Par exemple, aux minima régionaux du gradient de l'image.

prétraitement de l'image

La plupart des algorithmes de segmentation recommandent de traiter au préalable l'image à segmenter en lui appliquant un flou gaussien plus ou moins prononcé. Ce prétraitement se justifie en cela qu'il adoucit les bordures et réduit l'influence du bruit dans l'image, facilitant la segmentation. Empiriquement, pour des images aériennes, un léger flou gaussien suffit pour obtenir des segmentations en superpixels cohérentes. L'application de ce flou ne sert qu'à la segmentation, et peut bien entendu être abandonnée au moment de la classification.

La segmentation se fait dans la plupart des cas dans l'espace de couleurs CIELAB. Cet espace de couleurs est conçu pour refléter la vision humaine, en particulier la courbe de réponse de l'œil humain aux variations de couleurs, qui est logarithmique plutôt que linéaire. Cependant, cela nécessite de s'interroger sur la pertinence d'une telle conversion lorsque les trois canaux des images aériennes ne sont pas RVB, mais infra-rouge-rouge-vert (IRRV) par exemple. En pratique, cela ne semble pas poser de problèmes, mais ces techniques de segmentation ne se généraliseront donc pas nécessairement telles quelles pour des images dont la structure est différente du RVB traditionnel, en particulier pour le traitement d'images multispectrales. Seuls les algorithmes MRS et HSeg ont été conçus avec la télédétection comme application finale.

Forme et tailles des régions

Plusieurs analyses comparatives [47, 2, 59] ont permis d'établir les spécificités des principaux algorithmes de segmentation. La Figure 3.2 en illustre quelques exemples.

La principale source de variabilité entre différentes segmentations réside dans la géométrie des régions qu'elles produisent. La segmentation FH génère ainsi des régions dont la taille et la forme peuvent être fortement hétérogènes, car son exploration du graphe n'est pas contrainte. Ainsi, l'algorithme FH peut aussi bien regrouper des pixels similaires très éloignés dans une même région, tandis que d'autres ne comporteront que quelques pixels, sans qu'aucun paramètre ne permette de maîtriser cette variabilité. À l'opposé, les segmentations en superpixels, et plus particulièrement les dérivés de SLIC, produisent des régions visuellement régulières. Ces algorithmes possèdent un paramètre de compacité contrôlant l'adhérence à la grille sous-jacente. Les superpixels générés par SLIC peuvent ainsi aisément être contraints en taille en conservant une liberté de forme. Quickshift se comporte de manière similaire, bien que les superpixels générées soient nettement plus irréguliers que dans les méthodes dérivées de SLIC, ce qui produit des artefacts dans la segmentation. La segmentation *watershed* compacte présente des caractéristiques très similaires aux méthodes de superpixels, et il est préférable de l'utiliser tant l'approche classique est sensible au choix des marqueurs. Ces analyses sont conformes aux études de la littérature [47, 2]. Dans l'ensemble, de nombreuses variantes d'algorithme de superpixels ont été développées et présentent des caractéristiques similaires [59].

Cependant, nos travaux portent sur la classification d'images de télédétection et il est donc nécessaire d'étudier le comportement des algorithmes de segmentation sur des données prises au nadir. Un exemple est illustré dans la Figure 3.3. L'algorithme MRS semble particulièrement performant dans ce cadre. En effet, bien que la segmentation paraisse visuellement chaotique, la composition de l'image est respectée jusque dans les moindres détails. Les algorithmes de type superpixels tendent à faire disparaître les détails, et notamment les véhicules, ce qui peut poser problème pour les approches de modélisation d'objets.

Compte-tenu de cette analyse, nous étudierons en priorité les algorithmes de segmentation prévus pour la télédétection (HSeg et MRS) ainsi qu'un représentant des deux grandes familles de segmentation compacte : Quickshift et SLIC. Les approches *watershed* sont écartées compte-tenu de leur forte proximité avec SLIC [48] tandis que l'approche FH est éliminée de part sa grande variabilité entre régions [47]. Le choix des algorithmes de segmentation ayant été décidé, il s'agit désormais de s'atteler à l'extraction de caractéristiques.

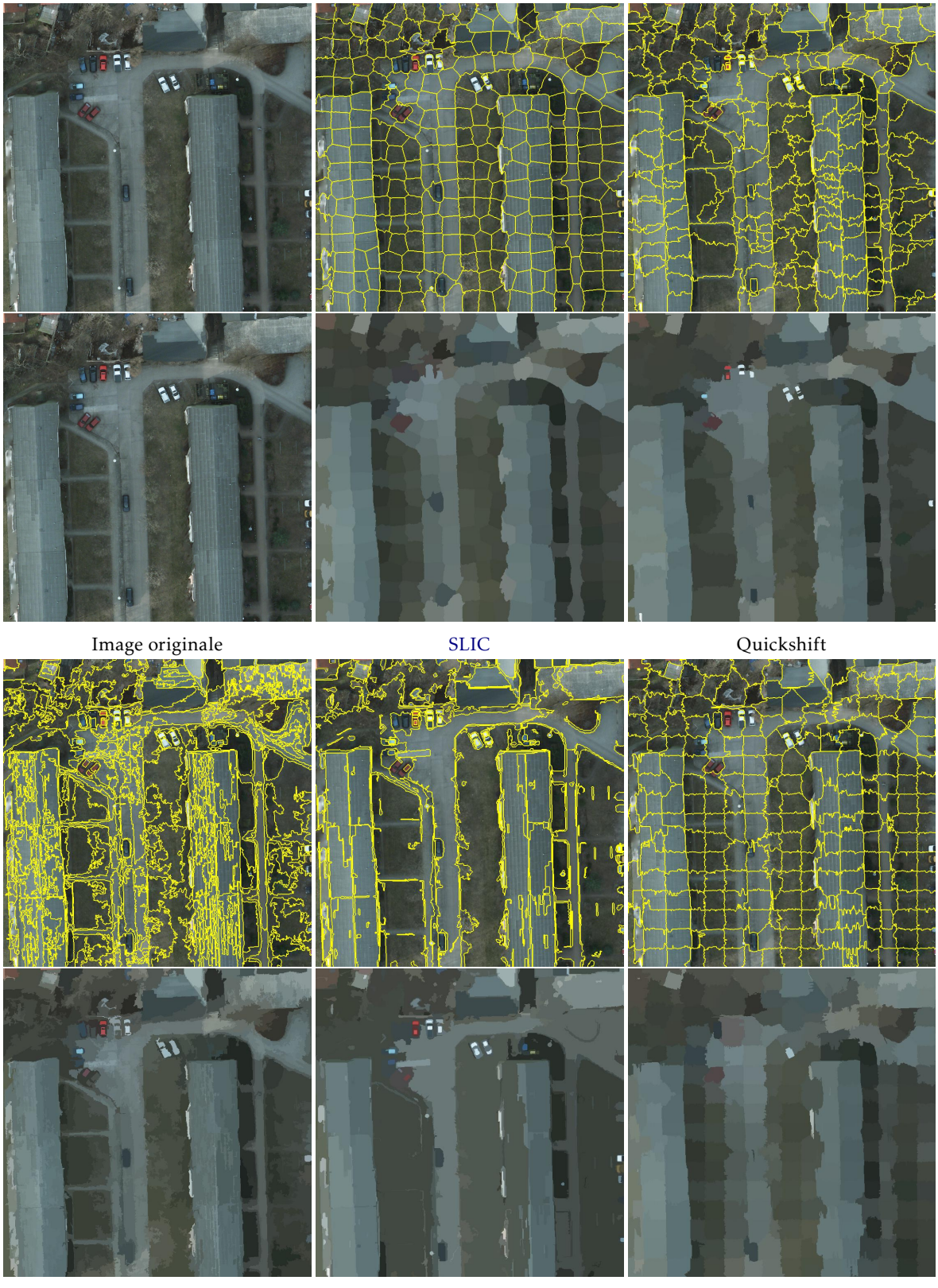


FIGURE 3.3 – Segmentations d'une image aérienne du jeu de données ISPRS Potsdam. Selon l'algorithme appliqué, les voitures sont plus ou moins bien segmentées.

3.2 Réseaux de neurones profonds

3.2.1 Réseaux de neurones convolutifs comme extracteurs de caractéristiques

L'intérêt majeur de l'apprentissage profond réside dans l'apprentissage des représentations [7, 24]. Nous avons vu au Chapitre 2 que les réseaux convolutifs réalisent une extraction de caractéristique apprise. Cette projection dans un espace de représentation est réalisée par les premières couches, qui sont elles-mêmes optimisables. Autrement dit, la représentation apprise est optimisée pour la tâche de classification sur les données d'entraînement.

Il est donc possible de fournir une image à un CNN et d'arrêter le calcul des activations avant la dernière couche. Les activations ainsi obtenues peuvent se représenter sous la forme d'un vecteur de caractéristiques.

Les caractéristiques ainsi extraites peuvent ensuite être utilisées pour entraîner un classifieur de façon habituelle. Cette approche est similaire au principe de spécialisation d'un réseau par *fine-tuning*. En particulier, il a été montré dans [55] que l'utilisation des caractéristiques extraites par un réseau préentraîné sur le jeu de données ImageNet [19] pour entraîner une SVM linéaire donnait d'excellents résultats sur la plupart des tâches visuelles. Razavian et al. [55] valident cette approche sur de nombreuses tâches et montrent qu'en dépit de sa simplicité de mise en œuvre, elle permet d'obtenir de meilleures performances qu'en utilisant les caractéristiques traditionnelles (HOG, SIFT…). Il est intéressant de constater que les représentations apprises par les réseaux convolutifs sont généralement de meilleurs points de départ pour l'optimisation que des initialisations aléatoires, même dans le cas de tâches très différentes [66].

Ces résultats ont été étendus à la classification d'images aériennes [52, 43, 31], de nombreux travaux ayant fait progresser l'état de l'art avec ces mêmes approches sur des jeux de données tels que *UC Merced* et *Brazilian Coffe*. En particulier, Marmanis et al. [43] et Penatti, Nogueira et dos Santos [52] ont montré qu'il est possible d'utiliser les caractéristiques extraites par un réseau préentraîné sur ImageNet pour la classification d'images aériennes et satellitaires, ce qui a été étendu à la segmentation sémantique par région par la suite [31]. Ce résultat est contre-intuitif dans la mesure où les images de la base ImageNet sont des images multimédia classiques : animaux, objets du quotidien, personnes, paysages… La généricité des filtres les rend néanmoins adaptables à de nombreux contextes, y compris la télédétection.

Application à la cartographie sémantique

À partir des procédés de segmentation et des méthodes de classification décrites précédemment, nous pouvons donc construire un processus complet de segmentation sémantique d'une image aérienne, repris de la Figure 3.4 :

1. Diviser l'image en sous-régions homogènes à l'aide d'un algorithme de segmentation.
2. Pour chaque région, extraire une pyramide d'images de dimensions 32×32, 64×64 et 128×128 autour du centroïde de la région pour intégrer différents niveaux de contexte spatial.
3. Extraire les caractéristiques de chaque imagette.
4. Concaténer les vecteurs résultants dans un unique vecteur de caractéristique.

Les échantillons d'apprentissage ainsi obtenus peuvent être utilisés pour entraîner le classifieur durant la phase d'apprentissage, ou simplement pour la prédiction en phase d'évaluation. Incidemment, l'étape de concaténation (4) permet également d'introduire des caractéristiques expertes ou multimodales [31] dans le processus d'apprentissage.

Dans le cas où l'extraction de caractéristiques est réalisée par un réseau convolutif, il est nécessaire de redimensionner l'imagette à la taille requise par le réseau (par exemple, 228×228 pour l'architecture AlexNet). Cette nécessité provient de la présence de couches

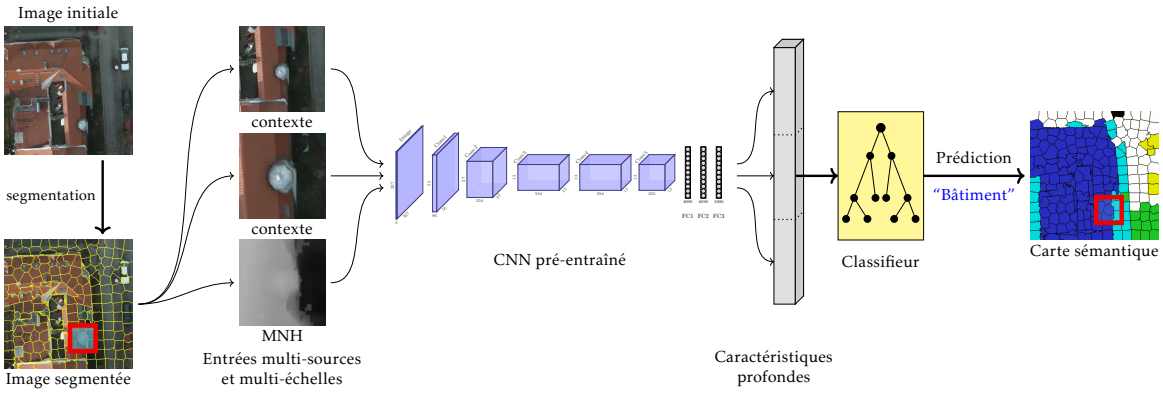


FIGURE 3.4 – Segmentation sémantique par régions d'une image aérienne. Chaque région de l'image segmentée est classifiée à partir de caractéristiques profondes extraites d'un réseau convolutif préentraîné.

entièrement connectées, qui contraignent la taille de la caractéristique obtenue en sortie de couches convolutives, et donc les dimensions initiales de l'image.

En outre, dans certains cas, la taille du vecteur de caractéristique est particulièrement grande. Pour AlexNet, la caractéristique obtenue est un vecteur de taille 1000 pour chaque imagette. Une région donnée génère donc un vecteur de taille 3000. L'optimisation exacte d'une SVM dans un tel espace étant extrêmement longue, nous utilisons l'approximation des vecteurs de support par descente de gradient de BOTTOU [10]. Les résultats de cette approche seront détaillés dans la Section 3.3.3.

3.2.2 Réseaux de neurones entièrement convolutifs

Les méthodes de classification par région présentent deux inconvénients majeurs. Tout d'abord, le niveau de détail de la carte sémantique finale est fortement limité par l'algorithme de segmentation utilisé. En effet, si l'algorithme produit des régions grossières, alors la carte sémantique le sera également, car la classification ne permet d'associer qu'une seule étiquette à chacune des régions traitées. Pour augmenter la résolution, il est alors nécessaire de produire des régions plus petites mais donc plus nombreuses, augmentant ainsi proportionnellement le temps de calcul nécessaire au traitement de l'intégralité de l'image. Dans le cas le plus extrême, la classification s'effectue directement sur les pixels (c'est-à-dire des régions de 1 px), les temps de calcul devenant prohibitifs sur des images de télédétection dont les dimensions dépassent le millier de pixels. Une solution envisageable consiste à utiliser les réseaux de neurones entièrement convolutifs. Comme présenté dans le Chapitre 2, les réseaux de neurones entièrement convolutifs, ou *Fully Convolutional Networks* (FCN), sont des réseaux comportant uniquement des couches de convolution conçus pour réaliser une classification dense. Ainsi, chaque pixel de l'image initiale peut se retrouver associée à une classe d'intérêt en une seule inférence.

Cette solution présente plusieurs avantages :

— La prédiction concernant un pixel prend automatiquement en compte le contexte spatial qui l'entoure,
— Les images d'entrée n'ont pas nécessairement une taille fixée *a priori*,
— La classification dense résultante est calée sur la grille des pixels, c'est-à-dire de la même résolution que l'image d'origine.

Les FCN peuvent ainsi être employés sur de grandes images en une seule passe, sans nécessiter de segmentation préalable. L'extraction de caractéristiques est automatiquement réalisée de façon dense, conjointement à la classification. Les représentations apprises pour la segmentation sémantique tiennent ainsi compte à la fois des propriétés colorimétriques des pixels et des relations spatiales existantes dans la cellule réceptive du FCN.

Application à la cartographie sémantique

De nombreuses architectures de FCN sont disponibles pour la segmentation sémantique. Nous retenons le modèle SegNet [6] (cf. Figure 3.5) qui présente un équilibre satisfaisant entre précision de la classification et temps de calcul. L'architecture de SegNet est symétrique et permet de replacer précisément les caractéristiques abstraites aux bonnes localisations spatiales. En outre, les résultats préliminaires avec les modèles FCN [42] et DeepLab [16] n'ont pas permis de constater d'améliorations significatives. Toutefois, nous soulignons que nos contributions ne sont pas spécifiques au modèle SegNet et peuvent être adaptées à n'importe quelle autre architecture. Nous comparons également SegNet au modèle ResNet-34 [28].

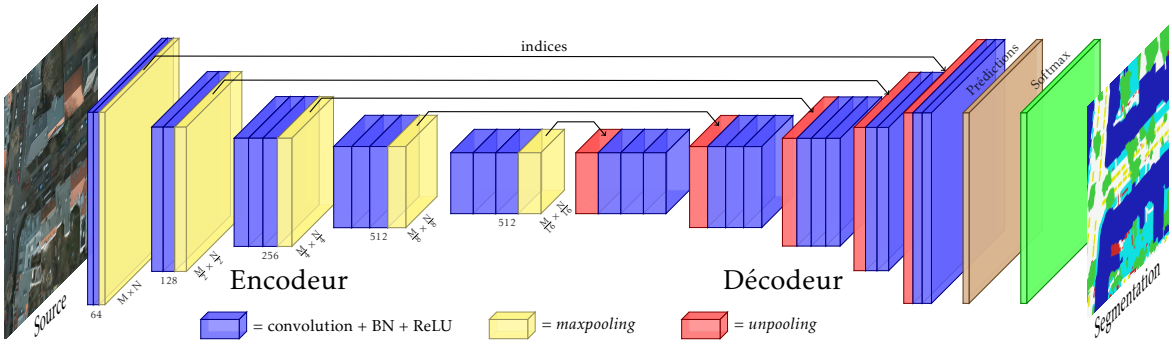


Figure 3.5 – Réseau de neurones entièrement convolutif – architecture SegNet [6].

SegNet est une architecture encodeur-décodeur conçue sur la base des couches convolutives du modèle VGG-16 [15, 58]. L'encodeur est une succession de couches élémentaires formées d'une convolution 3×3, d'une BN et d'une fonction de transfert ReLU. Chaque bloc de 2 ou 3 convolutions est suivi par une couche de sous-échantillonnage d'un facteur 2 et de pas égal à 2. L'architecture détaillée est représentée dans la Figure 3.5.

Le décodeur est une symétrie de l'encodeur et possède le même nombre de convolutions et le même nombre de blocs. Les réductions de dimensions sont remplacées par des sur-échantillonnages. Ceux-ci replacent les valeurs des activations intermédiaires aux indices ("*argmax*") calculés lors du sous-échantillonnage. Par exemple, la première couche de sous-échantillonnage calcule le masque des activations maximales et le transfère directement à la dernière couche de sur-échantillonnage. Les avant-dernières activations sont alors replacées aux positions ainsi transférées, le reste étant rempli par des zéros. Ces cartes d'activations éparses sont ensuite densifiées par les convolutions successives. Notons que pour que les opérations de sur-échantillonnage soient bien définies, il est nécessaire que la taille des images d'entrée soient une puissance de 2.

L'encodeur étant calqué sur VGG-16, ses poids sont initialisés à partir de ce même CNN préentraîné sur le jeu de données ImageNet [19]. Les poids du décodeur sont eux initialisés aléatoirement en utilisant la stratégie de He et al. [27]. On cherche alors à minimiser l'erreur empirique de classification moyenne sur chaque image, c'est-à-dire la moyenne de l'entropie croisée pour chaque pixel (i, j) entre son étiquette $y^{(i,j)}$ et les activations $z^{(i,j)}$ issues de SegNet, normalisées par un *softmax*. Si M et N désignent les dimensions de l'image d'entrée et k le nombre de classes, alors on cherche les poids du réseau minimisant :

$$\mathcal{L}(softmax(z), y) = -\frac{1}{\mathrm{M} \times \mathrm{N}} \sum_{i=1}^{\mathrm{M}} \sum_{j=1}^{\mathrm{N}} \sum_{p=1}^{k} y_p^{(i,j)} \log \left(\frac{\exp(z_p^{(i,j)})}{\sum_{q=1}^{k} \exp(z_q^{(i,j)})} \right). \tag{3.4}$$

Bien que les modèles entièrement convolutifs ne fixent pas les dimensions de l'image d'entrée, traiter les tuiles HR d'imagerie aérienne n'est pas réalisable compte-tenu de la

mémoire disponible sur les cartes graphiques actuelles. Par conséquent, nous adoptons une stratégie de contournement en traitant chaque tuile par sous-image.

Lors de l'apprentissage, des imagettes aléatoires sont extraites des tuiles disponibles. Dans une optique d'augmentation de données pour favoriser la capacité de généralisation du modèles, les images peuvent être aléatoirement transformées par symétrie horizontale ou verticale.

Lors de l'évaluation, les tuiles haute résolution sont traitées par fenêtre glissante. Pour limiter les effets de bord pouvant apparaître sur la grille, le pas de progression de la fenêtre glissante est inférieur aux dimensions de celle-ci. Cela génère ainsi un recouvrement, sur lequel nous pouvons moyenner les prédictions. Ce procédé permet de lisser les prédictions et d'améliorer les performances globales en réalisant plusieurs estimations pour le même pixel, aux dépens d'une légère augmentation du temps de calcul.

3.2.3 Aspects multiéchelles

Couche convolutive multinoyau

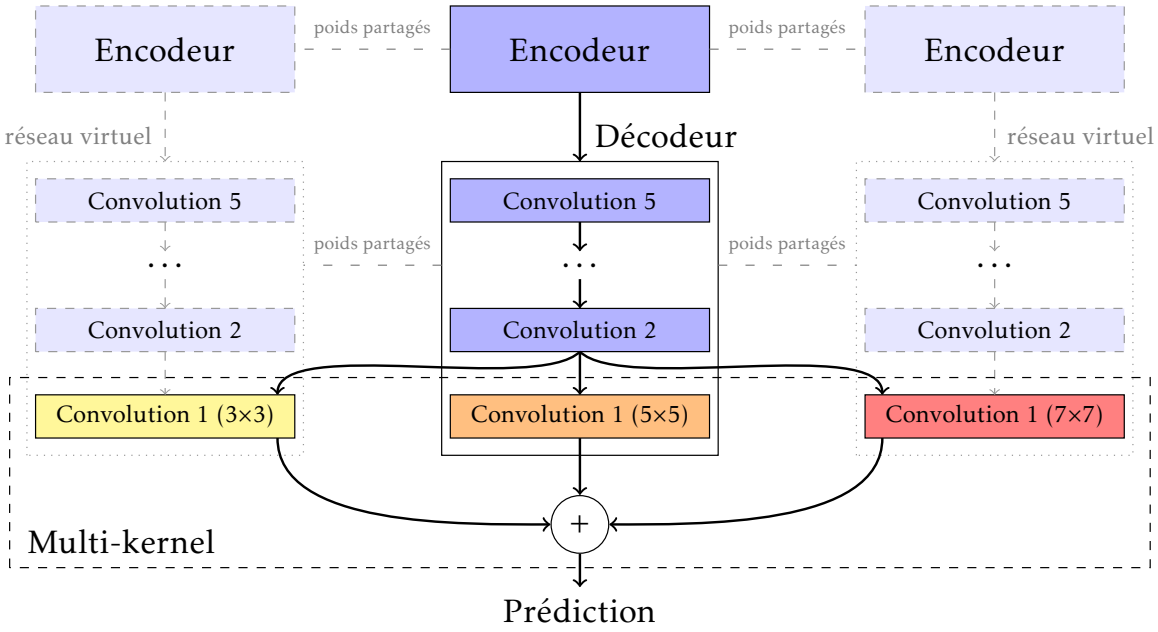


Figure 3.6 – Couche convolutive multinoyau. Une dernière couche convolutive opérant sur 3 voisinages spatiaux différents est équivalent à moyenner 3 modèles aux poids partagés.

Les approches convolutives multiéchelles ont montré à plusieurs reprises leur utilité pour la reconnaissance d'objets dans les réseaux Inception [60] et pour la segmentation sémantique [67], y compris en télédétection [68]. Nous proposons ici de modifier la dernière couche du décodeur de SegNet pour extraire plusieurs cartes d'activation prenant en compte différentes tailles de contexte spatial. En particulier, nous proposons d'utiliser non pas un unique noyau convolutif 3×3, mais un ensemble de convolutions 3×3, 5×5 et 7×7 opérant en parallèle. En pratique, ceci correspond à créer un ensemble de trois modèles partageant la même topologie et les mêmes poids, à l'exception de la dernière couche, comme illustré dans la Figure 3.6. En notant X_{in} les activations entrant dans la couche à plusieurs noyaux, $Z_p^{(s)}$ les activations en sortie à l'échelle s ($s \in [\![1, S]\!]$ avec ici $S = 3$ et $p \in [\![1, P]\!]$ avec P le nombre de plans de convolutions de l'avant-dernière couche, ici 64), Z_q^* les activations finales ($q \in [\![1, k]\!]$ avec k le nombre de classes) et $W_{p,q}^{(s)}$ le q^e noyau de convolution pour le p^e plan des activations à l'échelle s :

$$Z_q^* = \frac{1}{S}\sum_{s=1}^{S} Z_p^{(s)} = \frac{1}{S}\sum_{s=1}^{S}\sum_{p=1}^{P} W_{p,q}^{(s)} X_p \,. \tag{3.5}$$

Pour un pixel à la position (i,j) d'activation $z_k^{(s,i,j)}$ pour la classe k et l'échelle s, l'entropie croisée après *softmax* est obtenue par :

$$\mathcal{L}(softmax(z),y) = \sum_{l=1}^{k} y_l^{(i,j)} \log\left(\frac{\exp\left(\frac{1}{S}\sum_{s=1}^{S} z_l^{(s,i,j)}\right)}{\sum_{l'=1}^{k}\exp\left(\frac{1}{S}\sum_{s=1}^{S} z_{l'}^{(s,i,j)}\right)}\right). \tag{3.6}$$

S'il est possible d'entraîner le modèle en un seul bloc, il est toutefois plus flexible d'ajouter *a posteriori* des noyaux de convolution supplémentaires. Initialement, le réseau est entraîné sur une seule échelle. Après entraînement, il est possible de remplacer la dernière convolution par une autre avec un noyau plus petit ou plus grand, sur lequel on réalise un *fine-tuning*. Le noyau ainsi appris peut alors être ajouté à la dernière couche afin d'obtenir deux branches parallèles, et ainsi de suite.

Cette approche multinoyau se rapproche des blocs Inception [60] et de la convolution compétitive multiéchelle de Liao et Carneiro [36]. Cependant, ici seule la dernière couche comporte plusieurs noyaux convolutifs et le nombre de noyaux parallèles peut aisément être modifié après optimisation du modèle si la taille des objets d'intérêt vient à changer. Cette approche se retrouve dans le principe de l'agrégation de contextes de Yu et Koltun [67] utilisant des convolutions dilatées, permettant d'extraire des caractéristiques à plusieurs échelles. Toutefois, ici nous nous focalisons sur l'extraction de plusieurs tailles de contextes à une échelle locale et écartons la convolution à trous, coûteuse en temps de calcul, ou l'utilisation d'une pyramide d'images multiéchelle [68]. En comparaison, la méthode proposée ci-dessous est simple et flexible, et permet d'agréger des prédictions sur plusieurs contextes spatiaux à partir d'un extracteur de caractéristiques fixe.

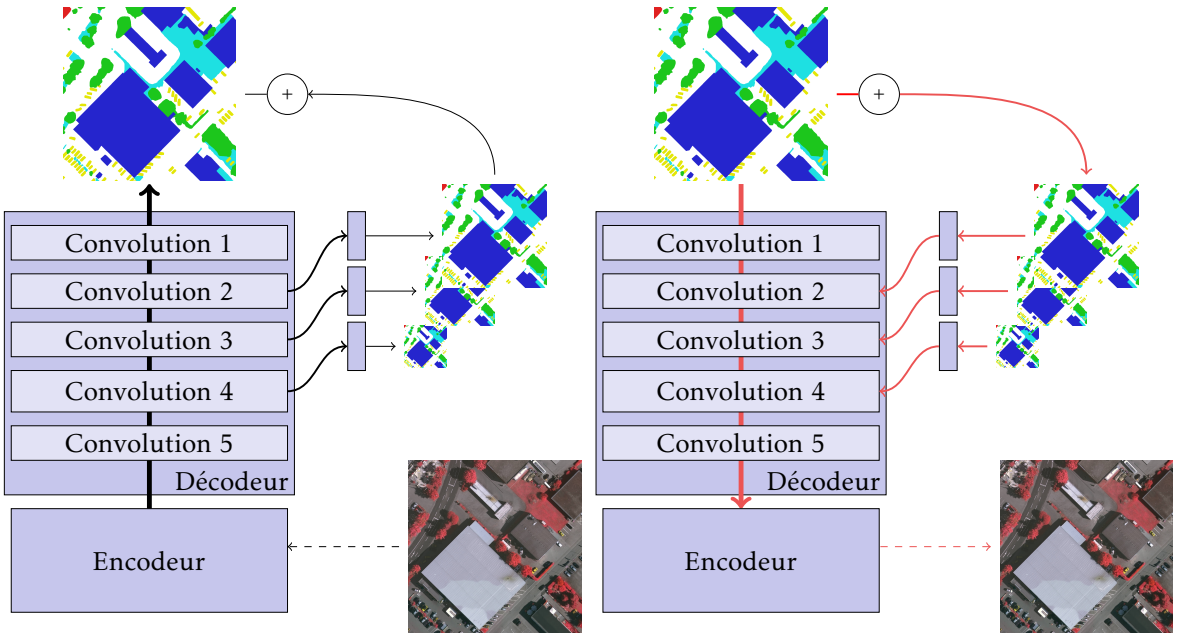
Supervision profonde

Le traitement multiéchelle des images de télédétection est généralement effectué en utilisant une approche pyramidale : différents contextes à différentes résolutions servent d'entrées à un ou plusieurs classifieurs. Nous proposons une approche alternative consistant à n'en traiter qu'une seule mais à produire en sortie du FCN une pyramide de prédictions, comme introduit dans le modèle DeepLab [16]. Chaque sortie est une carte prédite à une résolution différente sur laquelle il est possible de calculer une erreur qui sera rétropropagée dans le réseau. Ceci permet de réaliser d'une part une inférence multiéchelle et d'autre part d'introduire une forme de supervision profonde dans le modèle [32].

Dans le modèle SegNet, la pyramide de cartes d'activations apparaît naturellement dans le décodeur. Après le p^e bloc du décodeur, nous ajoutons une couche convolutive réalisant une classification à la résolution $\frac{2^p M}{32} \times \frac{2^p N}{32}$ (avec M, N les dimensions de l'image initiale I), comme illustré dans la Figure 3.7. Ces cartes sont ensuite interpolées à la résolution $M \times N$ et sommées pour obtenir la carte sémantique finale. En notant $P_{complète}$ la prédiction à pleine résolution, $P_{réduite_d}$ les cartes obtenues avec un facteur d'échelle $1:d$ et $\mathcal{I}_d$ l'interpolation bilinéaire d'un facteur d, la carte complète est obtenue par :

$$P_{complète} = \sum_{d\in\{0,2,4,8\}} \mathcal{I}_d(P_{réduite_d}) = P_0 + \mathcal{I}_2(P_2) + \mathcal{I}_4(P_4) + \mathcal{I}_8(P_8). \tag{3.7}$$

Lors de la rétropropagation, chaque bloc convolutif du décodeur reçoit deux gradients :
— Un gradient correspondant à la fonction de coût finale,



(a) Inférence multiéchelle sur un modèle SegNet. (b) Rétropropagation multiéchelle.

FIGURE 3.7 – Supervision profonde d'un SegNet à trois échelles.

— Un gradient correspondant à la fonction de coût réduite.

Les couches les plus profondes peuvent ainsi simplement apprendre à raffiner les prédictions de la couche précédente, ce qui simplifie l'optimisation globale du réseau [38].

3.3 Évaluation des modèles

Afin d'évaluer les performances relatives des différents modèles de segmentation et de classification introduits jusqu'ici, il est nécessaire de définir des critères quantitatifs autorisant la comparaison. Cette section détaille les métriques que nous utiliserons par la suite pour comparer différentes approches.

3.3.1 Métriques pour la classification

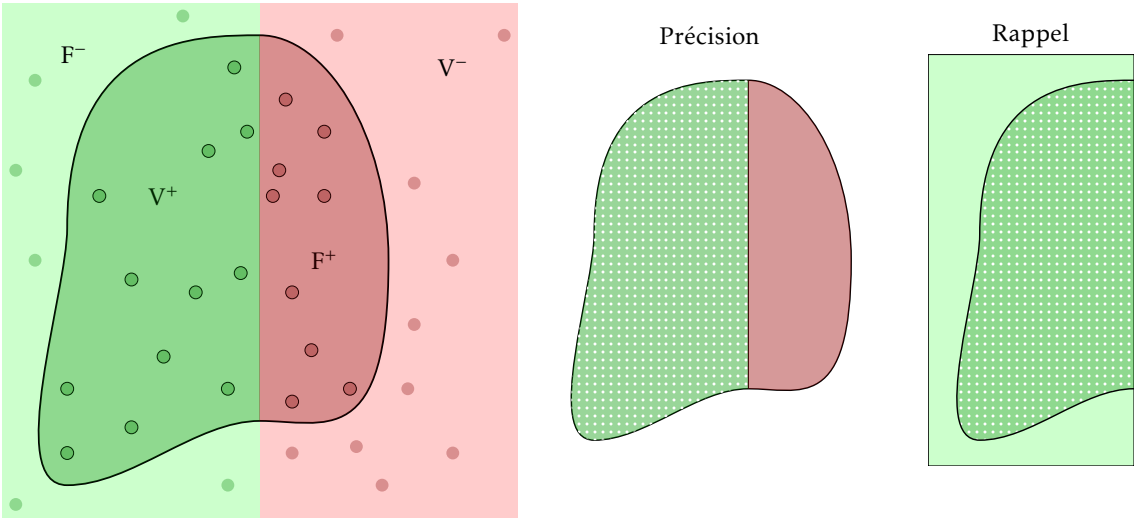


FIGURE 3.8 – Répartition des vrais positifs V^+, des vrais négatifs V^-, des faux positifs F^+ et des faux négatifs F^- pour une classification binaire dans un espace à deux dimensions.

Pour un classifieur donné et une classe d'intérêt i, on définit V^+ comme l'ensemble des vrais positifs (échantillons appartenant à la classe i correctement affectés), V^- l'ensemble des vrais négatifs (échantillons d'une classe $j \neq i$ n'étant pas affectés à i), F^+ l'ensemble des faux positifs (échantillons d'une classe $j \neq i$ ayant été affectés à i) et F^- l'ensemble des faux négatifs (échantillons de i affectés à $j \neq i$). Ce partionnement est illustré dans la Figure 3.8.

On définit alors les métriques de performance suivantes pour le classifieur, relativement à la classe i :

— La précision est définie comme le rapport entre le nombre de vrais positifs et le nombre total d'éléments affectés à la classe par le classifieur :

$$\textit{précision} = \frac{V^+}{V^+ + F^+}.$$

— Le rappel est défini comme le rapport entre le nombre de vrais positifs et le nombre total d'éléments appartenant réellement à la classe :

$$\textit{rappel} = \frac{V^+}{V^+ + F^-}.$$

— Le score F_1, ou coefficient de Sorensen-Dice, est défini comme la moyenne harmonique de la précision et du rappel :

$$F_1 = 2 \cdot \frac{\textit{précision} \times \textit{rappel}}{\textit{précision} + \textit{rappel}},$$

ce qui s'écrit également :

$$F_1 = \frac{2V^+}{2V^+ + F^+ + F^-}.$$

— L'exactitude est définie comme le rapport de prédictions exactes sur le nombre total d'échantillons :

$$exactitude = \frac{V^+ + V^-}{V^+ + F^+ + V^- + F^-} .$$

— L'intersection sur union (IsU), ou indice de Jaccard, est définie comme le rapport du nombre de prédictions exactes sur l'ensemble des prédictions de la classe et des échantillons réels :

$$IsU = \frac{V^+}{V^+ + F^+ + F^-} .$$

Le score F_1 et l'IsU ont l'avantage de ne pas être biaisés en faveur d'une classe dominante. Par exemple, un jeu de données contenant 95% de fond et 5% d'objet sera classifié à 95% d'exactitude par un classifieur prédisant systématiquement "fond". Cependant, le score F_1 de ce classifieur serait de 0.

L'IsU est proche du score F_1, mais accorde une pondération plus importante aux vrais positifs. Toutefois, les deux métriques peuvent être utilisées pour ordonner des classifieurs. Étant donné que $IsU/F = 1/2+IsU/2$, il existe une relation monotone entre les deux métriques. Un classifieur A meilleur qu'un classifieur B pour l'IsU le sera également pour le score F_1, et réciproquement.

Dans un cadre multiclasse, on s'intéressera à l'exactitude globale et à la moyenne de l'intersection sur union ou à la moyenne du score F_1 sur l'ensemble des classes. En complément, il sera également possible de s'appuyer sur le Kappa de Cohen mesurant la concordance entre les prédictions et la vérité terrain par rapport à un tirage aléatoire :

$$\kappa = \frac{P(accord) - P(hasard)}{1 - P(hasard)}$$

avec $P(accord)$ la proportion d'accord entre les prédictions et la vérité terrain et $P(hasard)$ la probabilité d'un accord aléatoire.

3.3.2 Métriques pour la segmentation

Dans un premier temps, il s'agit d'évaluer les capacités théoriques des différents algorithmes de présegmentation. En effet, si la segmentation rassemble dans une même région des pixels appartenant à deux classes différentes, il apparaîtra nécessairement des erreurs dans la classification finale, car une région ne sera associée qu'à une unique classe.

De fait, nous pouvons comparer les algorithmes de segmentation sur des images de référence selon quatre critères :

— L'erreur de sous-segmentation (ESS), définie comme le ratio de pixels appartenant à une région qui en recouvrent une autre. Formellement, en notant respectivement $\mathcal{S}$ et $\mathcal{R}$ la segmentation générée et la segmentation réelle, et N le nombre de pixels de l'image :

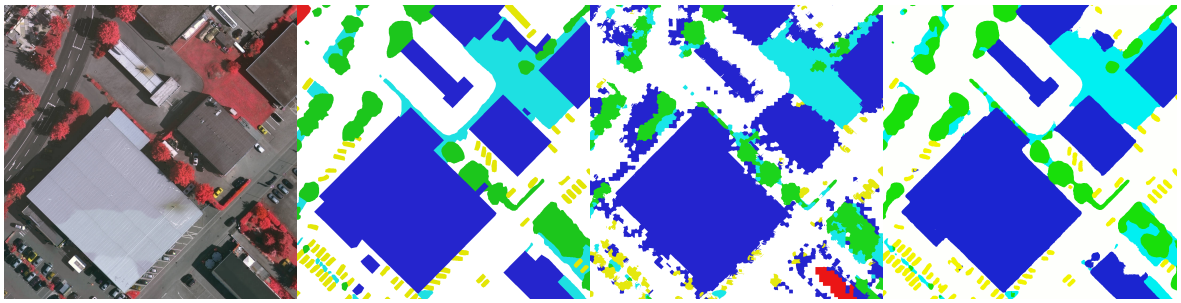
$$ESS = \frac{1}{N} \sum_{R_i \in \mathcal{R}} \sum_{S_j \in \mathcal{S}/S_j \cap R_i \neq \emptyset} \min(\left|R_i \cap S_j\right|, \left|R_i \backslash R_i \cap S_j\right|)$$

— Le rappel sur les bordures (RB), défini comme le rappel statistique des pixels placés à la frontière des segments qui se trouvent dans un 3-voisinage des frontières réelles :

$$RB = \frac{V^+}{V^+ + F^-}$$

— La pureté moyenne (PM), définie comme le pourcentage moyen de pixels d'une région appartenant à la classe localement la plus représentée. En notant maj l'opérateur qui, pour une région S_i renvoie la classe qui y est la plus représentée :

$$PM = \frac{1}{|\mathcal{S}|} \sum_{S_i \in \mathcal{S}} \frac{|\{p \in S_i \text{ et } classe(p) = maj(S_i)\}|}{|S_i|}$$



(a) Image IRRV (b) Vérité terrain (c) SLIC+AlexNet+SVM (d) SegNet

FIGURE 3.9 – Comparaison des cartes prédites en classification par régions et classification par FCN. Les prédictions denses de SegNet sont nettement plus précisees et visuellement plus détaillées que la carte obtenue par segmentation superpixels et caractéristiques profondes.
Légende : blanc : routes, bleu : bâtiments, cyan : végétation basse, vert : arbres, jaune : véhicules, rouge : autre.

— L'oracle, défini comme le taux de bonne classification pixellique qui serait obtenu par un classifieur parfait, assignant la classe majoritaire à chaque segment. Il s'agit de la meilleure classification possible théoriquement obtenable avec la segmentation considérée.

3.3.3 Classification par région

Nous choisissons d'évaluer différents algorithmes de segmentation non-supervisée dans un cadre de classification par région sur le jeu de données ISPRS 2D *Semantic Labeling* Vaihingen. Celui-ci comporte une acquisition aérienne EHR de la ville allemande de Vaihingen sur les canaux IRRV et est annoté pour 6 classes d'intérêt. Les propriétés du jeu de données sont détaillées dans l'Annexe A.1.1.

Nous comparons les algorithmes de segmentation les plus couramment utilisés dans la communauté vision par ordinateur et dans la communauté télédétection, identifiés dans la Section 3.1.2 : SLIC, LSC, Quickshift, MRS et HSeg. Les paramètres des segmentations sont réglés afin d'obtenir un nombre de régions similaire et les meilleures performances possibles. Ces algorithmes de segmentation sont représentatifs des différentes approches classiques de la littérature.

TABLEAU 3.1 – Comparaison des algorithmes de segmentation sur le jeu de données ISPRS Vaihingen. Les meilleurs résultats sont en **gras** et les suivants sont en *italique*.

Algorithme	Régions	ESS (%)	RB (%)	PM (%)	Oracle (%)
SLIC	≃**20 000**	**10,21**	84,07	75,10	*89,91*
LSC	≃22 800	*11,37*	91,13	71,54	85,83
Quickshift	≃*21 000*	11,66	88,34	72,90	83,61
MRS	≃23 500	13,12	**95,71**	**79,08**	**91,68**
HSeg	≃*21 000*	11,39	*94,83*	*78,66*	85,25

Nous appliquons ces algorithmes de segmentation sur l'ensemble des images du jeu de données ISPRS Vaihingen. Nous utilisons l'implémentation des auteurs pour LSC [35], l'implémentation de GUYET, MALINOWSKI et BENYOUNÈS [26] de l'algorithme MRS (adapté depuis la bibliothèque TerraLib [13]) et les implémentations de la bibliothèque scikit-image [63] pour SLIC et *Quickshift*. Les résultats sont détaillés dans le Tableau 3.1.

Concernant les métriques de segmentation pure, les algorithmes conçus pour le traitement d'images de télédétection excellent. En particulier, les algorithmes MRS et HSeg présentent un

Tableau 3.2 – Résultats de segmentation sémantique sur le jeu de validation ISPRS Vaihingen. Les meilleurs résultats sont en **gras** et les suivants sont en *italique*.

Algorithme	Régions	Exactitude (%)	Score F_1 (véhicules)	κ	Oracle (%)
SLIC	≃**20 000**	*82,20*	0,54	**0,76**	*89,91*
LSC	≃22 800	**82,45**	**0,58**	**0,76**	85,53
Quickshift	≃*21 000*	82,05	0,52	*0,75*	83,61
MRS	≃23 500	80,53	*0,56*	0,73	**91,68**
HSeg	≃*21 000*	79,56	0,54	0,72	85,25
Fenêtre glissante	≃23 800	81,22	0,53	0,74	92,56

rappel sur les bordures et une pureté moyenne élevés. Cela signifie que les frontières définies par ces segmentations sont proches des véritables régions sémantiques du jeu de données. Ceci n'est pas surprenant dans la mesure où les critères de similarité *ad hoc* qu'utilisent ces algorithmes sont spécifiquement conçus pour segmenter des objets de télédétection et sont donc particulièrement adaptés aux images du jeu de données ISPRS Vaihingen. Néanmoins, nous avons pu observer précédemment que ceci venait au prix de régions irrégulières, ce qui augmente l'erreur de sous-segmentation. En comparaison, les algorithmes de type superpixels sont moins performants car les régions sont de formes plus contraintes. L'accumulation de petites régions réparties sur l'image diminue l'erreur de sous-segmentation, mais introduit par ailleurs des superpixels moins purs et collant moins aux contours réels des objets. Dans l'ensemble, les performances théoriques de classification atteignables (oracle) varient de 83% à 91%. Les algorithmes MRS et SLIC semblent tirer leur épingle du jeu sur cette métrique.

Le Tableau 3.2 détaille les résultats obtenus après une classification par le protocole décrit précédemment. Le CNN AlexNet pour l'extraction de caractéristiques est implémenté en utilisant la bibliothèque d'apprentissage profond Caffe [29]. Le classifieur utilisé est une SVM linéaire optimisée par descente de gradient telle qu'implémentée dans la bibliothèque scikit-learn [51]. Il s'avère que le classement par taux de bonne classification ne correspond pas au classement utilisant l'oracle comme mesure. Cela signifie que les performances brutes de segmentation ne suffisent pas à déterminer la pertinence d'une segmentation dans un cadre d'extraction de caractéristiques.

En effet, les résultats de classification poussent à privilégier des approches de type superpixels. La régularité géométrique des segments bénéficie grandement au classifieur. Les segments présentent tous une compacité et une convexité forte. Au moment de l'extraction de l'imagette, la majorité des pixels au centre de l'image sont donc pertinents, et la caractéristique calculée par le réseau convolutif contiendra en grande partie de l'information issue de la région considérée. À l'inverse, les segmentations irrégulières s'imbriquent difficilement dans des imagettes rectangulaires, ce qui complexifie la tâche de classification car les échantillons d'apprentissage ne sont pas géométriquement normalisés, comme illustré par la Figure 3.3. En pratique, ces segmentations n'apportent aucun gain par rapport à une simple fenêtre glissante à coût calculatoire constant. La Figure 3.9 illustre un exemple de carte obtenue en appliquant l'algorithme SLIC et une classification par caractéristiques profondes. Si les grandes zones (bâtiments, routes, végétation) sont relativement bien délimitées, les bordures et les véhicules sont quant à eux très irréguliers.

Augmenter le paramètre de compacité de la segmentation MRS permet d'obtenir des régions plus homogènes semblables à des superpixels. Les résultats sont alors comparables à ceux obtenus avec SLIC, mais au prix d'une sursegmentation considérable : MRS nécessite deux fois plus de segments que SLIC pour obtenir la même précision. Ceci se répercute directement sur le temps de calcul, qui est proportionnel au nombre de régions à traiter. Enfin, remarquons que la détection des petits objets est sensible au choix de la présegmentation ; le score F_1 sur les véhicules peut ici être significativement amélioré par l'utilisation d'une

segmentation adaptée à des objets de petite taille, comme LSC.

3.3.4 Classification pixellique par segmentation sémantique

Comme nous l'avons vu, il apparaît clairement que la mise en œuvre d'une segmentation non-supervisée est le principal facteur limitant les performances des méthodes de classification par région. Non seulement l'utilisation de la segmentation introduit une borne supérieure aux performances de l'oracle, mais la géométrie même des exemples d'apprentissage qu'elle produit est peu adapté à l'extraction de caractéristiques profondes. L'étude des FCN capables d'apprendre de bout en bout la segmentation et la classification semble par conséquent particulièrement prometteuse.

Nous entraînons donc des modèles de réseaux profonds entièrement convolutifs SegNet et ResNet-34 sur les jeux de données ISPRS Vaihingen et ISPRS Potsdam. Nous traitons chaque tuile du jeu de données par une fenêtre glissante de dimensions 128×128 et un pas variable. Les modèles sont entraînés pendant 50 000 itérations avec une taille de *batch* de 10. Le taux d'apprentissage initial est fixé à 0,1 et est divisé par 10 après 35 000 et 45 000 itérations. Les réseaux sont implémentés à l'aide des bibliothèques Caffe [29] et PyTorch [53].

Dans un premier temps, nous validons cette approche uniquement sur les données pour lesquelles une vérité terrain est disponible, que nous divisons en deux sous-ensembles : apprentissage et validation. Pour comparer notre méthode à l'état-de-l'art, nous entraînons ensuite notre modèle sur l'ensemble du jeu de données (apprentissage + validation) avec les mêmes hyperparamètres. Nous soumettons enfin nos résultats sur le jeu de données de test au serveur d'évaluation de l'ISPRS, dont la vérité terrain nous est inconnue. Comme nous avons pu le constater dans la Figure 3.9, les prédictions pixelliques denses permettent d'obtenir des résultats visuellement prometteurs.

Recouvrement de la fenêtre glissante

Tableau 3.3 – Résultats de segmentation sémantique sur le jeu de validation ISPRS Vaihingen en fonction du recouvrement de la fenêtre glissante.

Modèle/Pas (px)	128	64	32
SegNet IRRV	87,8%	88,3%	88,8%
SegNet multinoyau	88,2%	88,6%	89,1%

L'utilisation d'une fenêtre glissante pour la segmentation de l'image pose la question du traitement des bordures. En effet, si le pas de la fenêtre glissante est identique aux dimensions de celle-ci, il risque alors d'apparaître des discontinuités aux bordures dégradant la qualité visuelles de la segmentation. En diminuant le pas, nous pouvons autoriser un recouvrement plus ou moins important entre deux fenêtres successives, c'est-à-dire que certains pixels pourront être observés à plusieurs reprises. Ceci augmente le temps d'inférence mais accroît également la précision du modèle, comme détaillé dans le Tableau 3.3. En effet, en divisant le pas par 2, le nombre d'imagettes à traiter est multiplié par 4. Cependant, moyenner plusieurs prédictions sur une même région permet de corriger des artefacts de classification, notamment le long des bords où le contexte spatial est manquant, et de lisser les discontinuités. L'expérience semble indiquer qu'un pas de 32px (75% de recouvrement) est suffisamment rapide pour la majorité des tâches et augmente significativement la précision (+1%). Une tuile complète est ainsi traitée en 4 minutes sur une NVIDIA Tesla K20c avec un pas de 32px et moins de 20 secondes avec un pas de 128px. Nous utiliserons donc ces paramètres pour la suite de nos travaux. Dans l'ensemble, le modèle SegNet parvient à correctement classifier plus de 87% des pixels du jeu de validation. En comparaison, aucune des méthodes de classification par région comparées précédemment ne dépassait 83%. Notamment, SegNet

parvient à dépasser les oracles sur les segmentations HSeg, LSC et *Quickshift*. Ceci montre la pertinence des réseaux entièrement convolutifs pour la segmentation sémantique : inférer une classification pixellique dense contraint SegNet à apprendre conjointement des caractéristiques prenant en compte les aspects spatiaux tout en respectant au mieux la résolution de l'image.

Transfert de connaissances

Tableau 3.4 – Comparaison de différentes initialisations sur le jeu de validation ISPRS Vaihingen.

Initialisation	Aléatoire	VGG-16 (ImageNet)			
Variabilité de l'encodeur $\frac{\alpha_e}{\alpha_d}$	1	1	0,5	0,1	0
Exactitude	87,0%	87,2%	**87,8%**	86,9%	86,5%

Le préentraînement d'un réseau profond sur un jeu de données générique est une pratique courante pour en augmenter les capacités de généralisation. ImageNet est ainsi souvent utilisé comme base de préentraînement pour la plupart des tâches visuelles. La télédétection ne fait pas exception, les filtres convolutifs appris sur des images multimédia pouvant être transférés pour la classification d'images aériennes [52]. Cependant, compte-tenu des différences importantes entre ces images et celles de télédétection, il peut exister un intérêt à laisser ces filtres précalculés évoluer librement lors de l'optimisation du réseau. Pour évaluer l'impact du préentraînement sur la classification d'images de télédétection, nous comparons différents taux d'apprentissage pour l'encodeur (α_e) et le décodeur (α_d) de SegNet. Nous testons notamment quatre stratégies :

— même variabilité : $\alpha_d = \alpha_e$,
— faible variabilité de l'encodeur : $\alpha_d = 2 \cdot \alpha_e$,
— très faible variabilité de l'encodeur : $\alpha_d = 10 \cdot \alpha_e$,
— gel de l'encodeur (pas de rétroprogation du gradient) : $\alpha_e = 0$.

Nous comparons ces résultats à ceux de référence obtenus avec une initialisation aléatoire de l'ensemble des paramètres du réseau [27], c'est-à-dire correspondant à un SegNet sans préentraînement et donc sans transfert de connaissances.

Comme détaillé dans le Tableau 3.4, le modèle réalise sa meilleure performance lorsque l'encodeur est initialisé à partir des poids préentraînés sur ImageNet et optimisé avec un taux d'apprentissage plus faible que celui du décodeur. Ceci renforce l'idée que des filtres convolutifs génériques donnent les meilleurs résultats lorsqu'il est possible de laisser l'optimisation les spécialiser sur une tâche particulière. Cependant, il est important de souligner qu'une variabilité trop grande induit un risque de surapprentissage. Ainsi, il est possible d'utiliser le taux d'apprentissage des paramètres préentraînés comme régularisation lors de l'optimisation. Ces résultats sont similaires aux conclusions de Nogueira, Penatti et dos Santos [49] et aux observations générales de Yosinski et al. [66] concernant le transfert de connaissances. Dans la suite de ces travaux, nous utiliserons donc les poids de VGG-16 préentraîné sur ImageNet lorsque cela est possible.

Choix du modèle

La comparaison en validation croisée entre les modèles SegNet et ResNet-34 ne permet pas de justifier l'utilisation d'un modèle résiduel aussi coûteux en mémoire. En effet, le Tableau 3.5 indique que les performances des deux modèles ne diffèrent que de 0,1% avec une variance légèrement plus faible concernant ResNet-34. Cependant, le ResNet-34 nécessite 25% de mémoire supplémentaire par rapport au modèle SegNet. En outre, le suréchantillonnage parcimonieux du SegNet lui permet d'être particulièrement précis pour la relocalisation de petits objets, comme les véhicules. Un ResNet plus profond, comme les

Tableau 3.5 – Résultats de segmentation sémantique en validation sur le jeu de données ISPRS Vaihingen.

Modèle	Routes	Bâtiments	Vég. basse	Arbres	Véhicules	Exactitude
SegNet	92,2± 2,1	95,6± 0,8	**82,6**± 4,2	**88,1**± 2,5	**88,2**± 0,6	90,2± 1,4
ResNet-34	**93,0**± 1,7	**96,0**± 0,6	82,3± 2,6	87,0± 3,7	87,0± 2,0	**90,3**± 1,0

ResNet-101, permettraient vraisemblablement d'extraire des caractéristiques plus puissantes que ResNet-34 et VGG-16. Cependant, compte-tenu de l'important surcoût calculatoire que cela engendrerait, nous travaillerons dans cette thèse principalement avec l'architecture SegNet.

Effets des approches multiéchelles

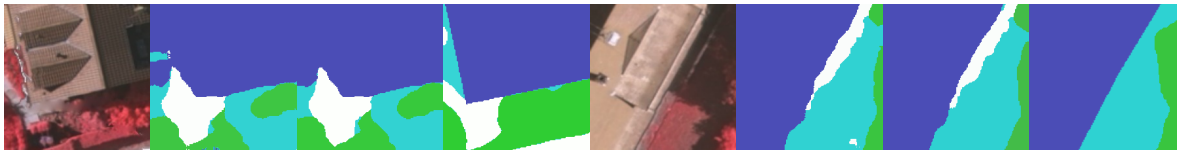


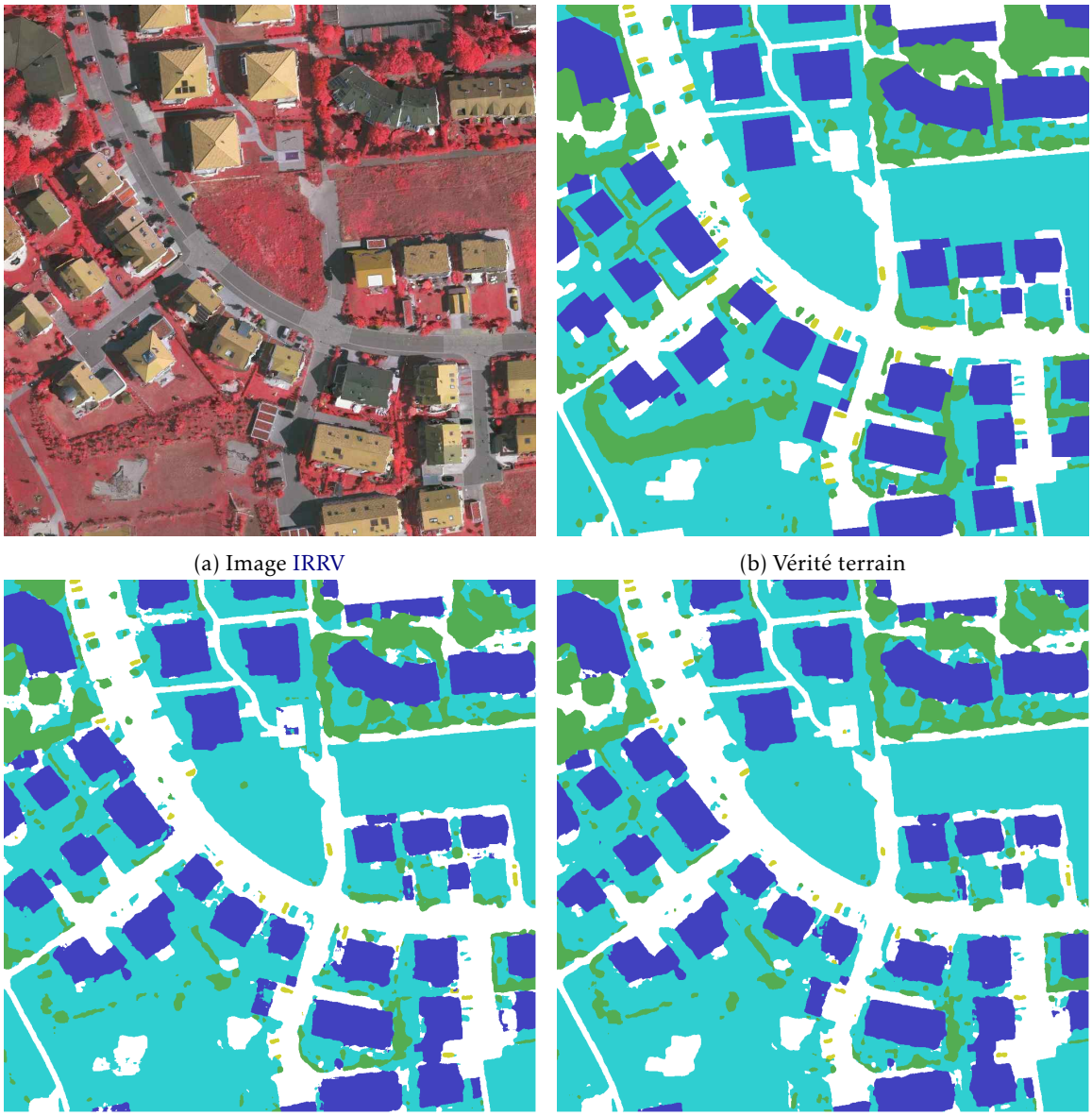
Figure 3.10 – Effets de la couche convolutive multinoyau sur des extraits du jeu de données ISPRS Vaihingen.
Légende : blanc : routes, bleu : bâtiments, cyan : végétation basse, vert : arbres, jaune : véhicules, rouge : autre.

Convolution multinoyau Comme indiqué dans le Tableau 3.3, l'utilisation d'une dernière couche convolutive multinoyau permet de gagner 0,4% d'exactitude supplémentaire. Ce gain accompagne un lissage des cartes prédites permettant de faire disparaître certains artefacts de classification sous forme de bruit poivre et sel illustré dans la Figure 3.10. Brahimi et al. [12] ont par la suite publié des résultats similaires en intégrant avec succès la convolution multinoyau dans un modèle DenseNet appliqué à la segmentation sémantique d'images de conduite autonome.

Tableau 3.6 – Résultats de validation multiéchelle sur le jeu de données ISPRS Vaihingen.

Nombre de branches	Routes	Bâtiments	Vég. basse	Arbres	Véhicules	Exactitude
Pas de branche	92,2	95,5	**82,6**	**88,1**	88,2	90,2± 1,4
1 branche	92,4	95,7	82,3	87,9	**88,5**	90,3± 1,5
2 branches	92,5	**95,8**	82,4	87,8	87,6	90,3± 1,4
3 branches	**92,7**	**95,8**	**82,6**	**88,1**	88,1	**90,5**± 1,5

Supervision profonde Le Tableau 3.6 permet d'identifier un apport positif léger de la supervision profonde multiéchelle sur SegNet sur les métriques considérées, avec un surcoût calculatoire quasiment nul. Comme attendu, les grandes structures bénéficient le plus des prédictions à faible échelle tandis que les voitures, les plus petits objets d'intérêt du jeu de données, sont plus difficiles à détecter à basse résolution. En outre, il semble que l'absence de structure dans la végétation induit également une confusion entre les arbres et la végétation basse aux échelles inférieures. Augmenter le nombre de branches n'augmente que marginale-



(a) Image IRRV (b) Vérité terrain

(c) SegNet (d) SegNet multiéchelle (3 branches)

FIGURE 3.11 – Effet de la supervision multiéchelle sur un extrait du jeu de données ISPRS Vaihingen. Les petits objets et les surfaces au contexte spatial ambigu bénéficient de la combinaison des prédictions à plusieurs échelles.
Légende : blanc : routes, bleu : bâtiments, cyan : végétation basse, vert : arbres, jaune : véhicules, rouge : autre.

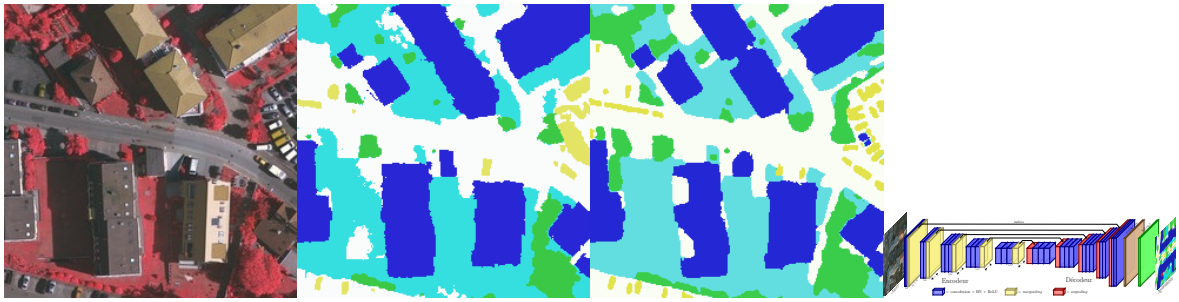
ment les performances du SegNet ce qui indique que la supervision profonde ne joue qu'un rôle limité par rapport à la fusion multiéchelle.

Bien que le gain quantitatif soit faible, une analyse visuelle des cartes obtenues après inférence montre que les améliorations qualitatives ne sont pas négligeables. Comme illustré dans la Figure 3.11, la prédiction multiéchelle permet de régulariser les prédictions et de réduire le bruit qui s'y trouve. Cela simplifie le traitement des cartes *a posteriori*, qu'il s'agisse de leur interprétation par un humain ou d'une vectorisation automatique. Ces résultats sont en adéquation avec les travaux postérieurs de JIANG et al. [30] pour la segmentation sémantique d'images RGB-D.

En outre, cette étude a également mis en avant que les cartes d'activation intermédiaires du décodeur sont quasiment aussi précises que les cartes à pleine résolution. Par exemple, la carte issue du deuxième bloc convolutif, c'est-à-dire à résolution 1 : 8 par rapport à l'image

initiale, est seulement 0,5% moins exacte que celle à résolution 1 : 1, l'essentiel des différences provenant de la classe "véhicule". Ceci était prévisible dans la mesure où les véhicules ne couvrent qu'environ 30 px en longueur à 9 cm/px, soit 3-4 px à résolution 1 : 8. Toutefois, les bonnes performances obtenues en utilisant uniquement les prédictions réduites indique qu'il serait possible de se limiter à un décodeur extrêmement simple comprenant uniquement un ou deux blocs convolutifs en décodeur en perdant peu de précision, soit une réduction du nombre de paramètres et du temps de calcul de SegNet d'environ 30%. Si la détection des petits objets n'est pas indispensable, stopper le calcul prématurément avec cette méthode permet d'accélérer significativement le temps d'inférence sur de nouvelles images.

Résultats finaux



(a) Image IRRV (b) RF + CRF [54] (c) "DLR" (FCN) [44] (d) **SegNet++**

FIGURE 3.12 – Comparaison des segmentations obtenues sur un extrait du jeu de test ISPRS Vaihingen. Légende : blanc : routes, bleu : bâtiments, cyan : végétation basse, vert : arbres, jaune : véhicules, rouge : autre.



Image IRRV Vérité terrain Prédiction

(a) Les cartes produites par SegNet sont parfois plus précises que la vérité terrain.



Image IRRV Vérité terrain Prédiction

(b) SegNet est susceptible de surapprendre sur des déformations géométriques dues à des erreurs d'orthorectification.

FIGURE 3.13 – Cas limites de désaccord entre les prédictions faites par SegNet et la vérité terrain. Légende : blanc : routes, bleu : bâtiments, cyan : végétation basse, vert : arbres, jaune : véhicules, rouge : autre.

Notre meilleur modèle améliore l'état-de-l'art sur le jeu de données ISPRS Vaihingen (cf. Tableau 3.7) [3]. La Figure 3.12 illustre une comparaison qualitative entre différentes méthodes. Les métriques sont calculées en ignorant un rayon de 3 pixels autour des bordures afin de tenir compte d'éventuelles imprécisions dans la vérité terrain. Au moment de la soumission de ces résultats, la meilleure méthode de l'état de l'art utilisait une combinaison de FCN et de caractéristiques expertes, tandis que la nôtre n'utilise que l'apprentissage statistique. La meilleure méthode précédente utilisant uniquement un FCN ("DLR_1") atteint 88,4%, ce que nous améliorons de 1%. Les précédentes méthodes utilisant les CNN atteignent 85,9% ("ONE_5"[11]) et 86,1% ("ADL_1"[50]). Notre méthode obtient des résultats supérieurs, sans recourir à des caractéristiques expertes ou à des post-traitement structurés comme les CRF.

Sur le jeu de données ISPRS Potsdam (cf. Tableau 3.8) [4], notre méthode est compétitive

3. http://www2.isprs.org/commissions/comm2/wg4/vaihingen-2d-semantic-labeling-contest.html
4. http://www2.isprs.org/commissions/comm2/wg4/potsdam-2d-semantic-labeling.html

Tableau 3.7 – Résultats du ISPRS 2D *Semantic Labeling Challenge* Vaihingen (ordre chronologique).

Méthode	Routes	Bâtiments	Vég. basse	Arbres	Véhicules	Exactitude
Stair Vision Library ("SVL_3") [21]	86,6	91,0	77,0	85,0	55,6	84,8
RF + CRF ("HUST") [54]	86,9	92,0	78,3	86,9	29,0	85,9
Ensemble de CNN ("ONE_5") [11]	87,8	92,0	77,8	86,2	50,7	85,9
FCN ("UZ_1") [65]	89,2	92,5	81,6	86,9	57,3	87,3
FCN ("UOA") [37]	89,8	92,1	80,4	88,2	82,0	87,6
CNN + MNH + RF + CRF ("ADL_3") [50]	89,5	93,2	82,3	88,2	63,3	88,0
FCN ("DLR_2") [44]	90,3	92,3	82,5	89,5	76,3	88,5
FCN + RF + CRF ("DST_2") [56]	90,5	93,7	83,4	89,2	72,6	89,1
SegNet++ (multinoyau) [4]	91,5	94,3	82,7	89,3	85,7	89,4
FCN + CRF + frontières + MNH corrigé ("DLR_9") [45]	92,4	95,2	83,9	89,9	81,2	90,3
ResNet-101 ("CASIA_2") [41]	93,2	96,0	84,7	89,9	86,7	91,1

avec l'état de l'art au moment de la soumission. Notamment, nous améliorons l'état de l'art sur les méthodes n'utilisant que la donnée optique de 0,3% par rapport au FCN de Sherrah [56] et de 4,2% par rapport au FCN de Volpi et Tuia [65]. Un exemple d'image complète segmentée est donné dans la Figure 3.14.

Il est intéressant de constater que les performances des modèles sont telles que certaines erreurs deviennent attribuables aux ambiguïtés des annotations. La Figure 3.13a illustre ainsi un cas où la vérité terrain ne suit pas parfaitement les contours de l'arbre, tandis que le modèle s'avère très fidèle. En outre, le processus d'orthorectification de la mosaïque d'images a introduit des distorsions géométriques qui ne sont pas prises en compte dans la vérité terrain, créant un désaccord entre l'apparence visuelle des pixels et la sémantique qui leur est attribuée, comme le montre la Figure 3.13b. Ces erreurs montrent par ailleurs qu'il devient difficile de faire progresser significativement les performances des modèles, tant les résultats

Tableau 3.8 – Résultats du ISPRS 2D *Semantic Labeling Challenge* Potsdam (ordre chronologique).

Méthode	Routes	Bâtiments	Vég. basse	Arbres	Véhicules	Exactitude
SVL [21]	83,5	91,7	72,2	63,2	62,2	77,8
FCN [56]	92,5	96,4	86,7	88,0	94,7	90,3
FCN + CRF + caractéristiques expertes [40]	91,2	94,6	85,1	85,1	92,8	88,4
FCN + CRF [65]	89,3	95,4	81,8	80,5	86,5	85,8
SegNet (IRRV)	92,4	95,8	86,7	87,4	95,1	90,0
ResNet-101 [41]	93,3	97,0	87,7	88,4	96,2	91,1

obtenus par les FCN sont proches de ce qui est raisonnablement attendu par les organisateurs du ISPRS 2D *Semantic Labeling Benchmark*. Le serveur d'évaluation a en effet été clos en juillet 2018, les performances plafonnant sur ces deux jeux de données.

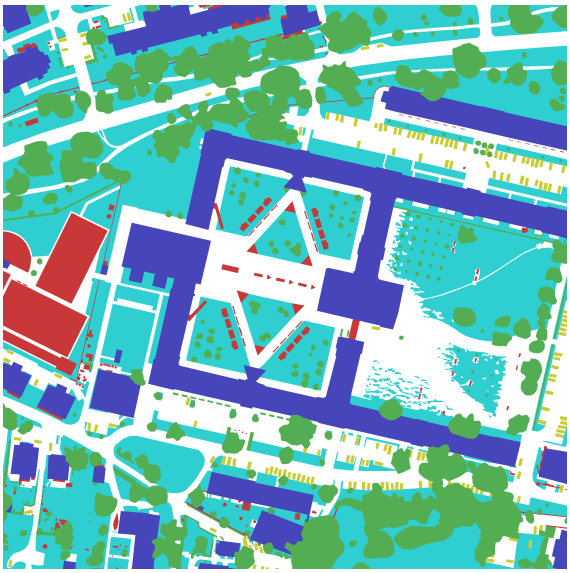
En conclusion, nous avons démontré que les FCN se prêtent particulièrement bien à la segmentation sémantique d'images aériennes. En particulier, sur la base de données ISPRS, nous avons pu montrer d'une part la nette supériorité des réseaux entièrement convolutifs par rapport aux approches de l'état de l'art en classification par région. En outre, nous avons proposé plusieurs bonnes pratiques concernant l'initialisation de ces réseaux et le paramétrage des fenêtres glissantes pour le traitement des images aériennes. Enfin, nous avons proposé deux méthodes de segmentation permettant d'inclure différentes échelles et contextes spatiaux au sein du réseau. Ces approches nous ont permis de faire progresser l'état de l'art. Toutefois, ces succès restent encore limités au domaine de l'imagerie 3 canaux IRRV et RVB à très haute résolution. Le chapitre suivant s'intéresse à étendre ces résultats sur d'autres capteurs couramment utilisés en observation de la Terre.

Les travaux présentés dans ce chapitre ont été le sujet de publications en conférence :

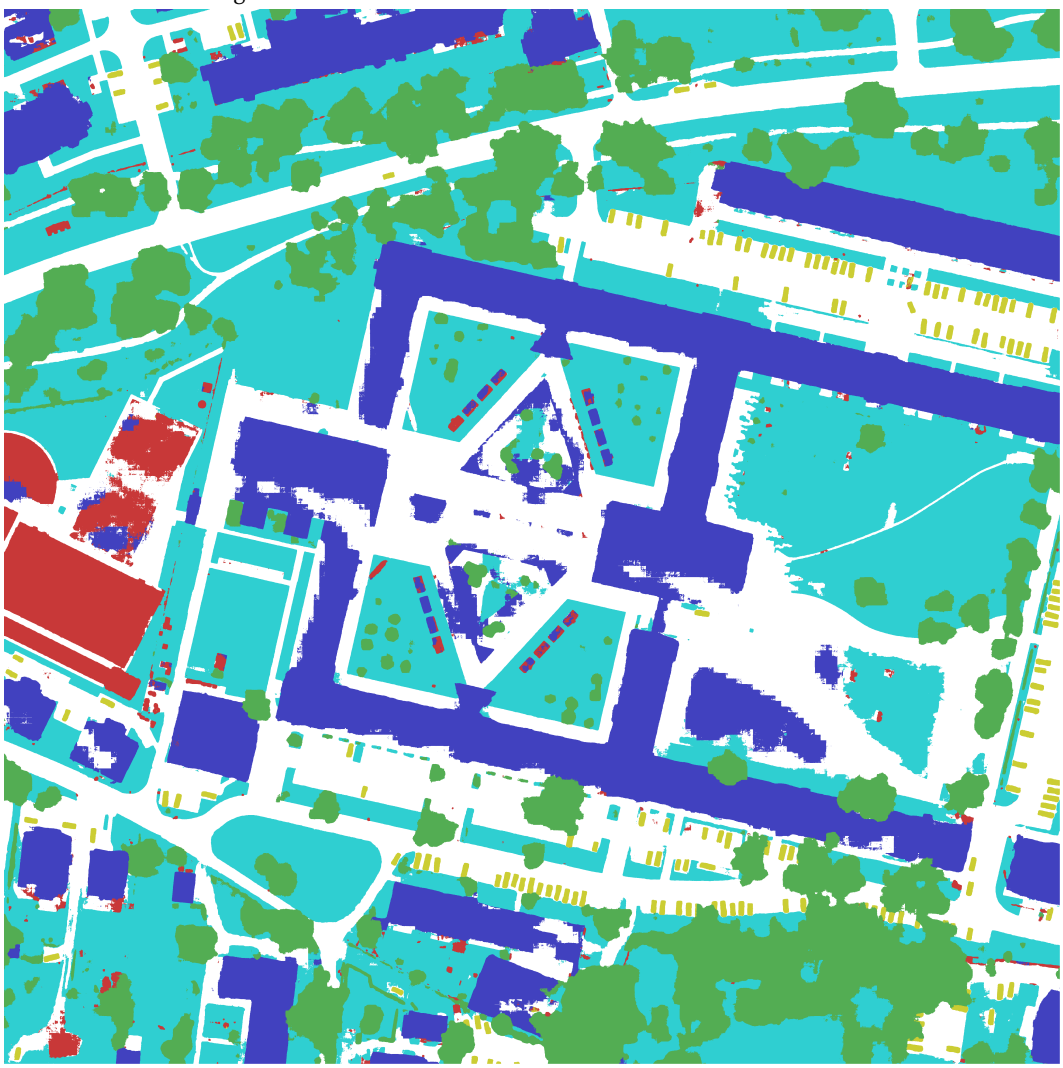
- Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « How Useful Is Region-Based Classification of Remote Sensing Images in a Deep Learning Framework ? » Dans : *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. Juil. 2016, p. 5091-5094. DOI : 10.1109/IGARSS.2016.7730327
- Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « Semantic Segmentation of Earth Observation Data Using Multimodal and Multi-Scale Deep Networks ». Dans : *Computer Vision – ACCV 2016*. Springer, Cham, 20 nov. 2016, p. 180-196. DOI : 10.1007/978-3-319-54181-5_12



Image RVB



Vérité terrain



Prédiction (SegNet)

Figure 3.14 – Carte sémantique obtenue par SegNet sur la tuile 3_11 du jeu de données ISPRS Potsdam Légende : blanc : routes, bleu : bâtiments, cyan : végétation basse, vert : arbres, jaune : véhicules, rouge : autre.

Références

[1] Radhakrishna Achanta et al. *SLIC Superpixels*. 2010. URL : http://infoscience.epfl.ch/record/149300 (cf. p. 62).

[2] Radhakrishna Achanta et al. « SLIC Superpixels Compared to State-of-the-Art Superpixel Methods ». Dans : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34.11 (nov. 2012), p. 2274-2282. ISSN : 0162-8828. DOI : 10.1109/TPAMI.2012.120 (cf. p. 64).

[3] Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « How Useful Is Region-Based Classification of Remote Sensing Images in a Deep Learning Framework ? » Dans : *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. Juil. 2016, p. 5091-5094. DOI : 10.1109/IGARSS.2016.7730327 (cf. p. 82).

[4] Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « Semantic Segmentation of Earth Observation Data Using Multimodal and Multi-Scale Deep Networks ». Dans : *Computer Vision – ACCV 2016*. Springer, Cham, 20 nov. 2016, p. 180-196. DOI : 10.1007/978-3-319-54181-5_12 (cf. p. 81, 82).

[5] Martin Baatz et Arno Schäpe. « Multiresolution Segmentation : An Optimization Approach for High Quality Multi-Scale Image Segmentation ». Dans : *Angewandte Geographische Informationsverarbeitung XII : Beiträge zum AGIT-Symposium Salzburg* (2000), p. 12-23 (cf. p. 63).

[6] Vijay Badrinarayanan, Alex Kendall et Roberto Cipolla. « SegNet : A Deep Convolutional Encoder-Decoder Architecture for Scene Segmentation ». Dans : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.12 (déc. 2017), p. 2481-2495. ISSN : 0162-8828. DOI : 10.1109/TPAMI.2016.2644615 (cf. p. 68).

[7] Yoshua Bengio, Aaron Courville et Pascal Vincent. « Representation Learning : A Review and New Perspectives ». Dans : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.8 (août 2013), p. 1798-1828. ISSN : 0162-8828. DOI : 10.1109/TPAMI.2013.50 (cf. p. 66).

[8] Serge Beucher et Fernand Meyer. « The Morphological Approach to Segmentation : The Watershed Transformation. Mathematical Morphology in Image Processing. » Dans : *Optical Engineering* 34 (1993), p. 433-481 (cf. p. 63).

[9] Petra Bosilj. « Indexation et recherche d'images par arbres des coupes ». Thèse de doct. Université de Bretagne Sud, 25 jan. 2016. URL : https://tel.archives-ouvertes.fr/tel-01362165/document (cf. p. 61).

[10] Léon Bottou. « Large-Scale Machine Learning with Stochastic Gradient Descent ». Dans : *In COMPSTAT*. 2010 (cf. p. 67).

[11] Alexandre Boulch. *DAG of Convolutional Networks for Semantic Labeling*. Office national d'études et de recherches aérospatiales, 2015. URL : https://www.itc.nl/external/ISPRS_WGIII4/ISPRSIII_4_Test_results/papers/onera_boulch.pdf (cf. p. 80, 81).

[12] Sourour Brahimi et al. « Multiscale Fully Convolutional DenseNet for Semantic Segmentation ». Dans : *WSCG 2018, International Conference on Computer Graphics, Visualization and Computer Vision*. Pilsen, Czech Republic, mai 2018. URL : https://hal.archives-ouvertes.fr/hal-01786688 (cf. p. 78).

[13] Gilberto Câmara et al. « TerraLib : An Open Source GIS Library for Large-Scale Environmental and Socio-Economic Applications ». Dans : *Open Source Approaches in Spatial Data Handling*. Advances in Geographic Information Science. Springer, Berlin, Heidelberg, 2008, p. 247-270. ISBN : 978-3-540-74830-4 978-3-540-74831-1. DOI : 10.1007/978-3-540-74831-1_12. URL : https://link.springer.com/chapter/10.1007/978-3-540-74831-1_12 (cf. p. 74).

[14] Tony Chan et Luminita Vese. « An Active Contour Model without Edges ». Dans : *Scale-Space Theories in Computer Vision*. International Conference on Scale-Space Theories in Computer Vision. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, 26 sept. 1999, p. 141-151. ISBN : 978-3-540-66498-7 978-3-540-48236-9. DOI : 10.1007/3-540-48236-9_13. URL : https://link.springer.com/chapter/10.1007/3-540-48236-9_13 (cf. p. 63).

[15] Ken Chatfield et al. « Return of the Devil in the Details : Delving Deep into Convolutional Nets ». Dans : *Proceedings of the British Machine Vision Conference*. British Machine Vision Conference (BMVC). British Machine Vision Association, 2014, p. 6.1-6.12. ISBN : 978-1-901725-52-0. DOI : 10.5244/C.28.6. URL : http://www.bmva.org/bmvc/2014/papers/paper054/index.html (cf. p. 68).

[16] Liang-Chieh Chen et al. « DeepLab : Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs ». Dans : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40.4 (avr. 2018), p. 834-848. ISSN : 0162-8828. DOI : 10.1109/TPAMI.2017.2699184 (cf. p. 68, 70).

[17] Dorin Comaniciu et Peter Meer. « Mean Shift : A Robust Approach toward Feature Space Analysis ». Dans : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24.5 (mai 2002), p. 603-619. ISSN : 0162-8828. DOI : 10.1109/34.1000236 (cf. p. 63).

[18] Camille Couprie et al. « Toward real-time indoor semantic segmentation using depth information ». Dans : *Journal of Machine Learning Research* (2014). ISSN : 1532-4435. URL : https://nyuscholars.nyu.edu/en/publications/toward-real-time-indoor-semantic-segmentation-using-depth-informa (cf. p. 61).

[19] Jia Deng et al. « ImageNet : A Large-Scale Hierarchical Image Database ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Juin 2009, p. 248-255. DOI : 10.1109/CVPR.2009.5206848 (cf. p. 66, 68).

[20] Pedro F. Felzenszwalb et Daniel P. Huttenlocher. « Efficient Graph-Based Image Segmentation ». Dans : *International Journal of Computer Vision* 59.2 (sept. 2004), p. 167-181. ISSN : 0920-5691, 1573-1405. DOI : 10.1023/B:VISI.0000022288.19776.77 (cf. p. 61).

[21] Markus Gerke. *Use of the Stair Vision Library within the ISPRS 2D Semantic Labeling Benchmark (Vaihingen)*. International Institute for Geo-Information Science and Earth Observation, 2015. URL : https://www.researchgate.net/profile/Markus_Gerke/publication/270104226_Use_of_the_Stair_Vision_Library_within_the_ISPRS_2D_Semantic_Labeling_Benchmark_(Vaihingen)/links/54ae59c50cf2828b29fcdf4b.pdf (cf. p. 81).

[22] Rémi Giraud, Vinh-Thong Ta et Nicolas Papadakis. « Robust Superpixels Using Color and Contour Features along Linear Path ». Dans : *Computer Vision and Image Understanding* (31 jan. 2018). ISSN : 1077-3142. DOI : 10.1016/j.cviu.2018.01.006. URL : http://www.sciencedirect.com/science/article/pii/S1077314218300067 (cf. p. 62).

[23] Maoguo Gong et al. « Superpixel-Based Difference Representation Learning for Change Detection in Multispectral Remote Sensing Images ». Dans : *IEEE Transactions on Geoscience and Remote Sensing* 55.5 (mai 2017), p. 2658-2673. ISSN : 0196-2892. DOI : 10.1109/TGRS.2017.2650198 (cf. p. 61).

[24] Ian Goodfellow, Yoshua Bengio et Aaron Courville. *Deep Learning*. MIT Press, 2016. URL : http://www.deeplearningbook.org (cf. p. 66).

[25] Leo Grady. « Random Walks for Image Segmentation ». Dans : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28.11 (2006), p. 1768-1783 (cf. p. 62).

[26] Thomas Guyet, Simon Malinowski et Mohand Cherif Benyounès. « Extraction des zones cohérentes par l'analyse spatio-temporelle d'images de télédétection ». Dans : *Revue Internationale de Géomatique* 25.4 (2015), p. 473-494. ISSN : 1260-5875, 2116-7060. DOI : 10.3166/RIG.25.473-494. URL : https://rig.revuesonline.com/articles/lvrig/abs/2015/04/lvrig254p473/lvrig254p473.html (cf. p. 74).

[27] Kaiming He et al. « Delving Deep into Rectifiers : Surpassing Human-Level Performance on ImageNet Classification ». Dans : *Proceedings of the IEEE International Conference on Computer Vision*. IEEE International Conference on Computer Vision (ICCV). Déc. 2015, p. 1026-1034. DOI : 10.1109/ICCV.2015.123 (cf. p. 68, 77).

[28] Kaiming He et al. « Deep Residual Learning for Image Recognition ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, United States, juin 2016, p. 770-778. DOI : 10.1109/CVPR.2016.90 (cf. p. 68).

[29] Yangqing Jia et al. « Caffe : Convolutional Architecture for Fast Feature Embedding ». Dans : *Proceedings of the 22Nd ACM International Conference on Multimedia*. MM '14. New York, NY, USA : ACM, 2014, p. 675-678. ISBN : 978-1-4503-3063-3. DOI : 10.1145/2647868.2654889. URL : http://doi.acm.org/10.1145/2647868.2654889 (cf. p. 75, 76).

[30] Jindong Jiang et al. « RedNet : Residual Encoder-Decoder Network for Indoor RGB-D Semantic Segmentation ». Dans : (4 juin 2018). arXiv : 1806.01054 [cs]. URL : http://arxiv.org/abs/1806.01054 (cf. p. 79).

[31] Adrien Lagrange et al. « Benchmarking Classification of Earth-Observation Data : From Learning Explicit Features to Convolutional Networks ». Dans : *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS). Juil. 2015, p. 4173-4176. DOI : 10.1109/IGARSS.2015.7326745 (cf. p. 66).

[32] Chen-Yu Lee et al. « Deeply-Supervised Nets ». Dans : *Artificial Intelligence and Statistics*. Artificial Intelligence and Statistics. 21 fév. 2015, p. 562-570. URL : http://proceedings.mlr.press/v38/lee15a.html (cf. p. 70).

[33] Alex Levinshtein et al. « TurboPixels : Fast Superpixels Using Geometric Flows ». Dans : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31.12 (déc. 2009), p. 2290-2297. ISSN : 0162-8828. DOI : 10.1109/TPAMI.2009.96 (cf. p. 63).

[34] Hongguang Li et al. « Superpixel-Based Feature for Aerial Image Scene Recognition ». Dans : *Sensors* 18.1 (8 jan. 2018), p. 156. DOI : 10.3390/s18010156. URL : http://www.mdpi.com/1424-8220/18/1/156 (cf. p. 61).

[35] Zhengqin Li et Jiansheng Chen. « Superpixel Segmentation Using Linear Spectral Clustering ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, p. 1356-1363. DOI : 10.1109/CVPR.2015.7298741. URL : http://www.cv-foundation.org/openaccess/content_cvpr_2015/html/Li_Superpixel_Segmentation_Using_2015_CVPR_paper.html (cf. p. 62, 74).

[36] Zhibin Liao et Gustavo Carneiro. « Competitive Multi-Scale Convolution ». Dans : (17 nov. 2015). arXiv : 1511.05635 [cs]. URL : http://arxiv.org/abs/1511.05635 (cf. p. 70).

[37] Guosheng Lin et al. « Efficient Piecewise Training of Deep Structured Models for Semantic Segmentation ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, United States, 2016, p. 3194-3203. DOI : 10.1109/CVPR.2016.348. URL : http://arxiv.org/abs/1504.01013 (cf. p. 81).

[38] Guosheng Lin et al. « RefineNet : Multi-Path Refinement Networks with Identity Mappings for High-Resolution Semantic Segmentation ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Juil. 2017, p. 5168-5177. DOI : 10.1109/CVPR.2017.549 (cf. p. 71).

[39] Ming-Yu Liu et al. « Entropy Rate Superpixel Segmentation ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. CVPR 2011. Juin 2011, p. 2097-2104. DOI : 10.1109/CVPR.2011.5995323 (cf. p. 62).

[40] Yansong Liu et al. « Dense Semantic Labeling of Very-High-Resolution Aerial Imagery and LiDAR with Fully-Convolutional Neural Networks and Higher-Order CRFs ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Honolulu, United States, juil. 2017, p. 1561-1570. DOI : 10.1109/CVPRW.2017.200 (cf. p. 81).

[41] Yongcheng Liu et al. « Semantic Labeling in Very High Resolution Images via a Self-Cascaded Convolutional Neural Network ». Dans : *ISPRS Journal of Photogrammetry and Remote Sensing* (21 déc. 2017). ISSN : 0924-2716. DOI : 10.1016/j.isprsjprs.2017.12.007. URL : http://www.sciencedirect.com/science/article/pii/S0924271617303854 (cf. p. 81).

[42] Jonathan Long, Evan Shelhamer et Trevor Darrell. « Fully Convolutional Networks for Semantic Segmentation ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Juin 2015, p. 3431-3440. DOI : 10.1109/CVPR.2015.7298965 (cf. p. 68).

[43] D. Marmanis et al. « Deep Learning Earth Observation Classification Using ImageNet Pretrained Networks ». Dans : *IEEE Geoscience and Remote Sensing Letters* 13.1 (jan. 2016), p. 105-109. ISSN : 1545-598X. DOI : 10.1109/LGRS.2015.2499239 (cf. p. 66).

[44] Dimitrios Marmanis et al. « Semantic Segmentation of Aerial Images with an Ensemble of CNNs ». Dans : *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences* 3 (1er juin 2016), p. 473-480. DOI : 10.5194/isprs-annals-III-3-473-2016. URL : http://adsabs.harvard.edu/abs/2016ISPAnIII3..473M (cf. p. 80, 81).

[45] Dimitrios Marmanis et al. « Classification With an Edge : Improving Semantic Image Segmentation with Boundary Detection ». Dans : *ISPRS Journal of Photogrammetry and Remote Sensing* (2017). DOI : 10.1016/j.isprsjprs.2017.11.009. arXiv : 1612.01337 (cf. p. 81).

[46] P. Márquez-Neila, L. Baumela et L. Alvarez. « A Morphological Approach to Curvature-Based Evolution of Curves and Surfaces ». Dans : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36.1 (jan. 2014), p. 2-17. ISSN : 0162-8828. DOI : 10.1109/TPAMI.2013.106 (cf. p. 63).

[47] Peer Neubert et Peter Protzel. « Superpixel Benchmark and Comparison ». Dans : *Proc. Forum Bildverarbeitung*. 2012, p. 1-12. URL : http://books.google.com/books?hl=en&lr=&id=39rFsOLJiRAC&oi=fnd&pg=PA205&dq=%22Liu+et+al.%22+%22algorithm+using+the+same+implementation,+data+set+and+error%22+%22image+edges+by+placing+them+inside+a+superpixel.+Depending%22+%22of+the+segmentation+compared+to+human+ground+truth%22+&ots=DmTz25PMw2&sig=TQoa_LmdyN4zyJIfJhugNHaSHEM (cf. p. 64).

[48] Peer Neubert et Peter Protzel. « Compact Watershed and Preemptive SLIC : On Improving Trade-Offs of Superpixel Segmentation Algorithms. » Dans : *ICPR*. 2014, p. 996-1001. URL : https://www.tu-chemnitz.de/etit/proaut/forschung/rsrc/cws_pSLIC_ICPR.pdf (cf. p. 62-64).

[49] Keiller Nogueira, Otávio Penatti et Jefersson A. dos Santos. « Towards Better Exploiting Convolutional Neural Networks for Remote Sensing Scene Classification ». Dans : (3 fév. 2016). arXiv : 1602.01517 [cs]. URL : http://arxiv.org/abs/1602.01517 (cf. p. 77).

[50] Sakrapee Paisitkriangkrai et al. « Effective Semantic Pixel Labelling with Convolutional Networks and Conditional Random Fields ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Juin 2015, p. 36-43. DOI : 10.1109/CVPRW.2015.7301381 (cf. p. 80, 81).

[51] Fabian Pedregosa et al. « Scikit-Learn : Machine Learning in Python ». Dans : *Journal of Machine Learning Research* 12 (Oct 2011), p. 2825-2830. ISSN : ISSN 1533-7928. URL : http://www.jmlr.org/papers/v12/pedregosa11a.html (cf. p. 75).

[52] Otávio Penatti, Keiller Nogueira et Jefersson A. dos Santos. « Do Deep Features Generalize from Everyday Objects to Remote Sensing and Aerial Scenes Domains ? » Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). Juin 2015, p. 44-51. DOI : 10.1109/CVPRW.2015.7301382 (cf. p. 66, 77).

[53] *PyTorch : Tensors and Dynamic Neural Networks in Python with Strong GPU Acceleration*. http ://pytorch.org/. 2016-. URL : http://pytorch.org/ (cf. p. 76).

[54] Nguyen Tien Quang et al. « An Efficient Framework for Pixel-Wise Building Segmentation from Aerial Images ». Dans : *Proceedings of the Sixth International Symposium on Information and Communication Technology*. International Symposium on Information and Communication Technology (SoICT). ACM, 2015, p. 43. URL : http://dl.acm.org/citation.cfm?id=2833272 (cf. p. 80, 81).

[55] Ali Sharif Razavian et al. « CNN Features Off-the-Shelf : An Astounding Baseline for Recognition ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Juin 2014, p. 512-519. DOI : 10.1109/CVPRW.2014.131 (cf. p. 66).

[56] Jamie Sherrah. « Fully Convolutional Networks for Dense Semantic Labelling of High-Resolution Aerial Imagery ». Dans : (8 juin 2016). arXiv : 1606.02585 [cs]. URL : http://arxiv.org/abs/1606.02585 (cf. p. 81).

[57] Jianbo Shi et Jitendra Malik. « Normalized Cuts and Image Segmentation ». Dans : *IEEE Trans. Pattern Anal. Mach. Intell.* 22.8 (août 2000), p. 888-905. ISSN : 0162-8828. DOI : 10.1109/34.868688. URL : http://dx.doi.org/10.1109/34.868688 (cf. p. 61).

[58] Karen Simonyan et Andrew Zisserman. « Very Deep Convolutional Networks for Large-Scale Image Recognition ». Dans : *Proceedings of the International Conference on Learning Representations (ICLR)*. Mai 2015. URL : http://arxiv.org/abs/1409.1556 (cf. p. 68).

[59] David Stutz, Alexander Hermans et Bastian Leibe. « Superpixels : An Evaluation of the State-of-the-Art ». Dans : *Computer Vision and Image Understanding* 166 (1er jan. 2018), p. 1-27. ISSN : 1077-3142. DOI : 10.1016/j.cviu.2017.03.007. URL : http://www.sciencedirect.com/science/article/pii/S1077314217300589 (cf. p. 64).

[60] Christian Szegedy et al. « Going Deeper with Convolutions ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Juin 2015, p. 1-9. DOI : 10.1109/CVPR.2015.7298594. URL : http://www.cv-foundation.org/openaccess/content_cvpr_2015/html/Szegedy_Going_Deeper_With_2015_CVPR_paper.html (cf. p. 69, 70).

[61] James C. Tilton et al. « Best Merge Region-Growing Segmentation With Integrated Nonadjacent Region Object Aggregation ». Dans : *IEEE Transactions on Geoscience and Remote Sensing* 50.11 (nov. 2012), p. 4454-4467. ISSN : 0196-2892, 1558-0644. DOI : 10.1109/TGRS.2012.2190079. URL : http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6182584 (cf. p. 63).

[62] Michael Van den Bergh et al. « SEEDS : Superpixels Extracted via Energy-Driven Sampling ». Dans : *Computer Vision – ECCV 2012*. Sous la dir. d'Andrew Fitzgibbon et al. Lecture Notes in Computer Science 7578. Springer Berlin Heidelberg, 7 oct. 2012, p. 13-26. ISBN : 978-3-642-33785-7 978-3-642-33786-4. DOI : 10.1007/978-3-642-33786-4_2. URL : http://link.springer.com/chapter/10.1007/978-3-642-33786-4_2 (cf. p. 63).

[63] Stéfan van der Walt et al. « Scikit-Image : Image Processing in Python ». Dans : *PeerJ* 2 (19 juin 2014), e453. ISSN : 2167-8359. DOI : 10.7717/peerj.453. URL : https://peerj.com/articles/453 (cf. p. 74).

[64] Andrea Vedaldi et Stefano Soatto. « Quick Shift and Kernel Methods for Mode Seeking ». Dans : *Computer Vision – ECCV 2008*. Sous la dir. de David Forsyth, Philip Torr et Andrew Zisserman. Lecture Notes in Computer Science 5305. Springer Berlin Heidelberg, 12 oct. 2008, p. 705-718. ISBN : 978-3-540-88692-1 978-3-540-88693-8. DOI : 10.1007/978-3-540-88693-8_52. URL : http://link.springer.com/chapter/10.1007/978-3-540-88693-8_52 (cf. p. 63).

[65] Michele Volpi et Devis Tuia. « Dense Semantic Labeling of Subdecimeter Resolution Images With Convolutional Neural Networks ». Dans : *IEEE Transactions on Geoscience and Remote Sensing* 55.2 (fév. 2017), p. 881-893. ISSN : 0196-2892. DOI : 10.1109/TGRS.2016.2616585 (cf. p. 81).

[66] Jason Yosinski et al. « How Transferable Are Features in Deep Neural Networks ? » Dans : *Advances in Neural Information Processing Systems*. Neural Information Processing Systems (NIPS). 2014, p. 3320-3328. URL : http://papers.nips.cc/paper/5347-how-transferable-are-features-in-deep-neural-networks (cf. p. 66, 77).

[67] Fisher Yu et Vladlen Koltun. « Multi-Scale Context Aggregation by Dilated Convolutions ». Dans : *Proceedings of the International Conference on Learning Representations (ICLR)*. 23 nov. 2015. URL : http://arxiv.org/abs/1511.07122 (cf. p. 69, 70).

[68] Wenzhi Zhao et Shihong Du. « Learning Multiscale and Deep Representations for Classifying Remotely Sensed Imagery ». Dans : *ISPRS Journal of Photogrammetry and Remote Sensing* 113 (mar. 2016), p. 155-165. ISSN : 0924-2716. DOI : 10.1016/j.isprsjprs.2016.01.004. URL : http://www.sciencedirect.com/science/article/pii/S0924271616000137 (cf. p. 69, 70).

Chapitre 4

Extension aux capteurs non-conventionnels

I suppose it is tempting, if the only tool you have is a hammer, to treat everything as if it were a nail.

— Abraham Maslow

Sommaire

4.1 Images multispectrales 92
4.1.1 Prise en compte du proche infrarouge 92
4.1.2 Images multispectrales 94
4.2 Imagerie hyperspectrale 98
4.2.1 Principes physiques de l'imagerie hyperspectrale 98
4.2.2 Jeux de données 100
4.2.3 Approches traditionnelles 102
4.2.4 Apprentissage profond et imagerie hyperspectrale 104
4.3 Imagerie laser et modèles de terrain 109
4.3.1 Modèle de terrain 109
4.3.2 Construction d'une image composite 111

Résumé du chapitre :

Les images multispectrales sont couramment utilisées en télédétection, mais leur nombre de bandes élevé complique l'application directe de FCN préentraînés sur des données RVB. Dans ce chapitre, nous montrons qu'il est en pratique tout de même possible d'étendre au cas multispectral les résultats obtenus en imagerie couleur. Nous étudions tout d'abord le cas d'images aériennes IRRVB, avant d'adapter des FCN classiques pour la cartographie sémantique d'images multispectrales Sentinel-2, pour lesquelles la prise en compte des bandes non-visibles est largement bénéfique.

Nous nous intéressons ensuite au cas particulier de l'imagerie hyperspectrale, pour laquelle le nombre élevé de longueurs d'onde d'acquisition nécessite des précautions supplémentaires. Nous explorons les architectures des réseaux de l'état de l'art en apprentissage profond pour l'imagerie hyperspectrale et montrons qu'il existe un bénéfice significatif à déployer des modèles convolutifs à trois dimensions, en dépit du faible nombre d'exemples.

Enfin, nous étudions les possibilités de traitement des modèles numériques de terrain à partir de réseaux convolutifs. En effet, les modèles de terrain contiennent une information de hauteur particulièrement discriminante pour la végétation et les objets artificiels, absente des images orthorectifiées. Nous montrons que les FCN en niveaux de gris permettent d'obtenir des cartographies à partir de ces modèles, mais moins précises qu'en utilisant les images couleur, justifiant ainsi le besoin d'approches multimodales.

4.1 Images multispectrales

Le chapitre précédent nous a permis de constater les excellentes performances des FCN pour la segmentation sémantique d'images aériennes RVB et IRRV. Toutefois, les satellites d'observation de la Terre, qu'ils soient institutionnels (Landsat, SPOT…) ou commerciaux (IKONOS, WorldView…), embarquent en grande majorité des capteurs multispectraux. Ces instruments permettent d'observer de l'information invisible pour l'œil humain dont il est désirable de tirer profit pour la cartographie automatisée.

4.1.1 Prise en compte du proche infrarouge

Les capteurs multispectraux les plus simples réalisent simultanément une acquisition couleur et infrarouge, formant ainsi des images à 4 canaux IRRVB. De nombreux satellites d'observation de la Terre en sont équipés, comme SPOT et Pléiades en France. En outre, ces satellites effectuent également une acquisition dite panchromatique dont la résolution spatiale est nettement supérieure. Cette combinaison est courante car les méthodes de super-résolution (*pansharpening*) permettent de générer des images multispectrales à la résolution de l'image panchromatique. En première approche, nous pouvons considérer les images IRRVB du jeu de données ISPRS Potsdam qui présentent des propriétés similaires à celles attendues sur ces satellites.

Dans le Chapitre 3, nous avons établi que les FCN peuvent s'appliquer indifféremment aux images IRRV et RVB classiques. En particulier, nous avons été en mesure de transférer les poids de réseaux de préentraînés sur ImageNet à des images de télédétection IRRV. Toutefois, ce transfert devient impossible dans le cas des images multispectrales IRRVB, car la différence de nombre de canaux ne permet pas de conserver la topologie du réseau. Initialement, nous avons choisi de contourner ce problème en omettant le canal infrarouge et en ne considérant que les canaux visibles.

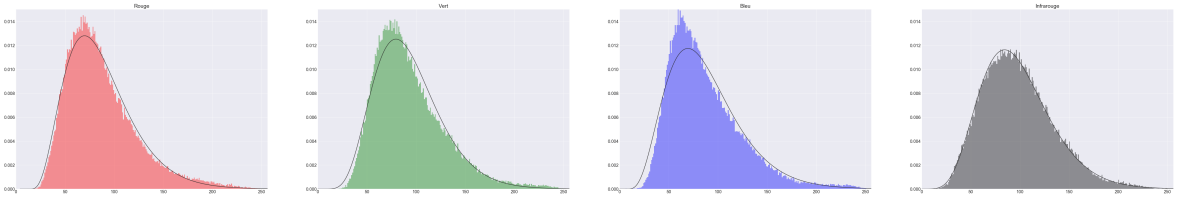


Figure 4.1 – Distributions des intensités pour les canaux rouge, vert, bleu et infrarouge du jeu de données ISPRS Potsdam.

Avant toute expérimentation, nous pouvons examiner les distributions des intensités pour chaque canal au sein du jeu de données. Les histogrammes présentés dans la Figure 4.1 révèlent que les distributions correspondent à des lois gamma dont les paramètres pour les canaux RVB sont très proches. Néanmoins, les statistiques du canal infrarouge dévient significativement des canaux visibles, ce phénomène étant mis en évidence par les cartes de corrélation inter-canaux de la Figure 4.2. Les canaux visibles sont ainsi fortement corrélés (coefficient de Pearson supérieur à 0,87). À l'inverse, le canal infrarouge n'est que modérément corrélé avec les autres canaux, notamment lorsque l'écart en longueur d'onde s'accroît. Ainsi, le coefficient de Pearson entre les canaux rouge et infrarouge est de 0,80, mais descend à 0,69 entre vert et infrarouge et 0,57 entre bleu et infrarouge.

Nous réalisons une étude préliminaire sur le jeu de données ISPRS Potsdam en reprenant les modèles et hyperparamètres décrits dans le Chapitre 3. En particulier, nous générons plusieurs variantes de SegNet, avec et sans initialisation de l'encodeur à partir des poids de VGG-16 préentraîné sur ImageNet, pour différentes combinaisons de canaux. Cela nous permet d'isoler d'une part les variations de performances dues aux bandes spectrales et d'autre part celles provoquées par le transfert de connaissances depuis ImageNet. Les tuiles

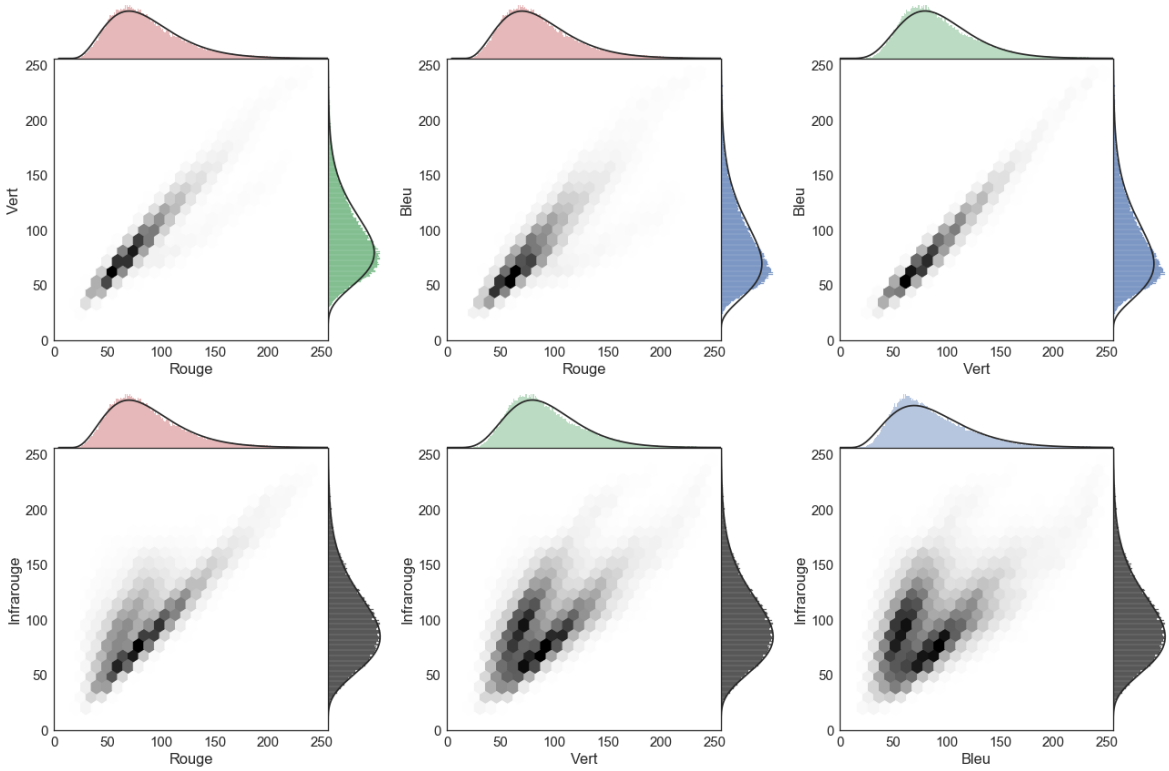


FIGURE 4.2 – Cartes de corrélation entre canaux du jeu de données ISPRS Potsdam.

2_10, 2_11, 2_12, 3_10, 3_11, 3_12, 4_10, 4_12, 5_11, 5_12, 6_7, 6_8, 6_9, 6_10, 6_11, 7_9, 7_10 et 7_12 sont utilisées pour l'entraînement tandis que la validation s'effectue sur les tuiles 4_11, 5_10, 6_12, 7_7, 7_8 et 7_11. Les résultats de cette expérience sont compilés dans le Tableau 4.1.

En premier lieu, il apparaît que le choix des 3 canaux RVB ou IRRV n'a que peu d'influence sur les performances de classification du modèle. Cependant, la prise en compte de l'infrarouge au sein d'un modèle à quatre canaux IRRVB dégrade significativement les résultats finaux. Ceci se vérifie également lorsque les poids des convolutions sont initialisés aléatoirement, c'est-à-dire sans préentraînement. Les résultats de classification des modèles entraînés sur 2 canaux seulement font apparaître une interaction négative entre les canaux infrarouge et bleu, entraînant une baisse conséquente des performances du modèle. Il semble ainsi y avoir un lien entre la cohérence radiométrique des canaux utilisés en entrée du réseau et la qualité finale de la classification. Ce lien est toutefois encore hypothétique et n'est pas

TABLEAU 4.1 – Comparaison des performances de segmentation sémantique de SegNet sur le jeu de données de validation ISPRS Potsdam pour différentes combinaisons de canaux. Le transfert correspond à une initialisation à partir des poids entraînés sur ImageNet.

Canaux	Transfert	Routes	Bâtiments	Vég. basse	Arbres	Véhicules	Autres	Exactitude
IR+B	✗	72,79	87,22	57,61	71,74	87,05	13,15	70,69
R+G	✗	89,92	95,80	81,49	84,30	95,21	40,42	88,48
IR+R	✗	90,75	95,89	82,77	84,97	95,50	42,47	89,25
RVB	✗	90,40	96,32	82,38	83,78	95,42	39,97	89,02
IRRV	✗	90,83	95,91	83,31	84,26	94,99	43,74	89,29
IRRVB	✗	89,67	95,57	82,35	83,82	95,17	42,89	88,51
RVB	✓	92,35	**97,62**	85,18	87,19	**96,11**	**52,15**	91,22
IRRV	✓	**92,51**	97,34	**85,68**	**87,54**	**96,11**	50,24	**91,28**

trivial à mettre en évidence.

4.1.2 Images multispectrales

La majorité des capteurs optiques embarqués dans des satellites d'observation de la Terre opèrent en multispectral. En effet, l'information intéressante ne se situe pas toujours dans le domaine visible. La réponse lumineuse de la chlorophylle dans l'infrarouge proche est par exemple un indicateur caractéristique de la végétation invisible pour l'humain. Les longueurs d'onde d'acquisition supplémentaires permettent ainsi d'identifier des matériaux spécifiques qu'il est impossible de détecter autrement. L'instrumentation de Sentinel-2 est conçue pour pouvoir détecter aussi bien les aérosols côtiers (bande 1 à 443 nm), que le *red edge* de la chlorophylle (bandes 5 à 7 entre 705 nm et 783 nm), la vapeur d'eau (bande 9 à 945 nm) et les cirrus (bande 10 à 1,375 µm). Les capteurs comportent, de par leur conception, une part de connaissance experte qu'il serait fâcheux d'ignorer.

Cette section s'intéresse donc à l'utilisation des FCN pour la segmentation sémantique d'images satellites multispectrales. Pour ce faire, nous tirons parti d'un ensemble d'images Sentinel-2 accompagné des cartes d'occupation des sols *GlobeCover* 2009 [1] fourni par Ben Hamida et al. [6]. Les images Sentinel-2 considérées couvrent une région se situant à la frontière de la France, de la Suisse et de l'Italie et ont été acquises entre mai et octobre 2016. Nous faisons l'hypothèse que les changements d'occupation des sols durant les 7 années séparant la création de la carte et l'acquisition des données sont peu nombreux relativement à la surface couverte.

Un premier jeu de données est constitué à partir des images de mai à octobre ne contenant aucune couverture nuageuse (nuages opaques ou cirrus). Le second jeu de données comporte l'ensemble des images, y compris en présence de nuages, mais est restreint temporellement à la période estivale (juin à août). Dans ce deuxième cas, les nuages sont considérés comme une classe additionnelle à segmenter, à partir du masque fourni par le programme Copernicus. Les nuages sont en effet un problème majeur pour les capteurs optiques, la lumière ne pouvant les traverser sans une atténuation significative. Leur prise en compte est un enjeu considérable car il s'agit d'une occlusion souvent présente dans la plupart des zones climatiques. Les deux jeux de données comportent ainsi différentes acquisitions sur les mêmes zones à plusieurs dates. Les tuiles Sentinel-2 sont interpolées à une résolution de 20 m/px pour l'ensemble des bandes. Les annotations issues de *GlobeCover* sont conservées à leur résolution originale de 300 m/px. Les Tableaux 4.2 et 4.3 récapitulent les propriétés des deux jeux de données et la liste des classes considérées.

Tableau 4.2 – Descriptifs des deux jeux de données d'images Sentinel-2 considérés. Le premier s'étend sur une longue période mais exclut les images contenant une couverture nuageuse. Le second est restreint à une période temporelle plus courte mais inclut les images en présence de nuages. Les deux jeux de données contiennent environ 150 millions de pixels chacun.

Jeux de données (période)	nombre d'images		
	entraînement	validation	nombre de classes
D1, période longue sans nuage (mai–oct. 2016)	140	54	16
D2, période courte avec nuages (juin–août 2016)	158	39	17

Afin de réaliser la segmentation sémantique sur ces images, nous considérons une architecture SegNet réduite, dont le décodeur est coupé après le deuxième bloc convolutif. En effet, il ne nous est pas utile d'obtenir des cartes à résolution 1 : 1. Nous utilisons donc l'approche multiéchelle de la Section 3.2.3 pour générer des cartes à résolution 1 : 8 (160 m/px), 1 : 16 et 1 : 32. Ceci diminue à la fois le temps de calcul et le nombre de paramètres à optimiser. Les

Tableau 4.3 – Liste des classes des jeux de données D1 et D2 dérivées de *GlobeCover* 2009. * La classe nuage est ajoutée *a posteriori* à partir du masque fourni avec les données Sentinel-2 par Copernicus.

Valeur	Couleur	Légende *GlobeCover* 2009
1		Cultures non irriguées
2		Mosaïque culture (50-70%)/végétation (pelouse, fruticée, forêt) (20-50%)
3		Mosaïque végétation (pelouse, fruticée, forêt) (50-70%)/cultures (20-50%)
4		Forêt décidue à feuilles larges dense (>40%, >5m)
5		Forêt épineuse sempervirent dense (>40%, >5m)
6		Forêt peu dense (15-40%) décidue ou sempervirent (>5m)
7		Forêt mixe décidue et sempervirent peu dense (>15%, >5m)
8		Mosaïque forêt et fruticée (50-70%)/pelouse (20-50%)
9		Mosaïque pelouse (50-70%)/forêt et fruticée (20-50%)
10		Fruticée peu dense (>15%, <5m)
11		Pelouse peu dense (>15%)
12		Végétation éparse (boisée, fruticées, pelouse, >15%)
13		Végétation peu dense (boisée, fruticées, pelouse) sur sol régulièrement inondé ou détrempé (eau douce, saumâtre ou salée)
14		Surface artificielle ou assimilée (urbanisation>50%)
15		Zone de terre nue
16		Étendue d'eau
17		Nuage*

cartes sont finalement interpolées à 300 m/px puis moyennées avant le calcul du *softmax*. La prédiction finale est comparée à la vérité terrain durant l'entraînement en utilisant l'entropie croisée.

Nous comparons pour cette tâche deux variantes de SegNet. La première, SegNet RVB, n'opère que sur les bandes 2,3 et 4 de Sentinel-2 correspondant à une image en vraies couleurs. La deuxième, SegNet MSI, est entraînée sur l'ensemble des 12 bandes [1]. Les hyperparamètres restants sont repris du Chapitre 3. Les réseaux sont optimisés jusqu'à convergence par descente de gradient stochastique avec moment, avec un taux d'apprentissage fixé à 0,001 et un moment de 0,9 pendant 150 000 itérations. Les variantes RVB et MSI utilisent respectivement une taille de *batch* de 20 et 10, pour chacune occuper environ 6 Go de mémoire GPU. L'entraînement prend environ 18 heures sur un GPU NVIDIA Titan X (génération Pascal) à l'aide de notre implémentation PyTorch [47]. Les paramètres sont tous initialisés aléatoirement, le nombre de bandes n'étant pas compatible avec une initialisation depuis VGG-16. [2]

Le modèle entraîné sur les 12 bandes atteint 66,5% d'exactitude sur le jeu de données D1 (sans nuage) et 86,4% sur D2 (avec nuages). L'écart important entre les jeux de données est une combinaison de deux facteurs. Tout d'abord, la présence des nuages nombreux et faciles à détecter (score $F_1 > 97\%$) augmente statistiquement le nombre de pixels bien classifiés. Néanmoins, comme le montre le Tableau 4.4, les performances sur l'ensemble des classes de D1 sont inférieures à celles des mêmes classes sur D2. Ceci s'explique par la faible variabilité des images de D1. Ce jeu de données ayant été constitué sous contrainte d'une couverture nuageuse faible, son étendue spatiale est plus faible et les conditions environnementales sont moins variées. Les modèles appris sur D1 généralisent donc moins bien sur les nouvelles acquisitions. À l'inverse, D2 couvre une période temporelle faible mais avec de nombreuses

1. La bande 8A n'est pas utilisée.
2. Par souci d'exhausivité, signalons cependant que des travaux préliminaires de Mace et al. [38] se sont intéressés à la duplication de filtres afin d'initialiser des réseaux en multispectral à partir de poids RVB obtenus sur ImageNet.

variations environnementales et donc une variété d'images plus importante. Cela permet au modèle d'apprendre les invariants radiométriques nécessaires à une bonne généralisation. Enfin, la prise en compte de l'ensemble des 12 bandes multispectrales permet de gagner 2% d'exactitude sur D1 et 2,5% sur D2 par rapport au modèle RVB seul. Ce phénomène n'est pas lié à une classe en particulier, puisque la majorité d'entre elles bénéficient des bandes spectrales additionnelles (cf. Tableau 4.4). Ceci conforte notre intuition initiale : l'information multispectrale est plus riche et plus discriminante que l'image couleur.

Il nous paraît indispensable de souligner que les annotations considérées proviennent d'une source ancienne (2009) et grossière (300 m/px). Cela explique notamment l'apparence pixellisée de la vérité terrain sur les illustrations, qui introduit une approximation dans le protocole d'évaluation. En effet, les scènes observées par Sentinel-2 peuvent avoir changé de type d'occupation des sols depuis 2009. En outre, la résolution de Sentinel-2 permet d'identifier des objets et des structures qui étaient mélangées, et donc invisibles, dans *GlobeCover*. Toutefois, l'adéquation géométrique entre les prédictions et la vérité terrain est visuellement forte. Il est plausible que certains désaccords entre le modèle et les annotations soient en réalité dus à la nature approximative de *GlobeCover* et aux changements temporels, et que les prédictions faites par SegNet soient en réalité plus précises sur certaines classes – notamment les surfaces artificialisées – que la vérité terrain dont nous disposons. Les Figures 4.3 et 4.4 illustrent quelques exemples qualitatifs de cartes générées par nos modèles.

Finalement, cette étude nous permet d'aboutir à deux conclusions. D'une part, elle montre la pertinence des modèles entièrement convolutifs pour le traitement des images multispectrales. En effet, l'architecture SegNet s'étend avec succès aux données Sentinel-2. S'il est nécessaire d'arbitrer certains choix techniques, notamment concernant l'interpolation des bandes et le choix d'une résolution de référence pour la vérité terrain, il ne semble pas y avoir de verrou majeur à l'utilisation des FCN pour les données multispectrales. D'autre part, l'inclusion des bandes hors du domaine visible permet d'augmenter l'expressivité du modèle, qui bénéficie de la richesse de l'information multispectrale. Cela permet notamment d'améliorer le pouvoir discriminant du SegNet sur des classes pouvant être ambiguës en RVB.

Tableau 4.4 – Performances de SegNet sur les jeux de données D1 et D2 Sentinel-2 (cf. Tableau 4.3 pour le détail des classes).

Jeu de données	Modèle	Exactitude	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
D1	SegNet RVB	55,0	39,0	35,5	1,81	71,0	0,91	0,00	0,00	0,00	0,00	2,08	0,00	5,33	0,00	36,0	0,00	74,6	–
	SegNet MSI	66,5	36,2	38,1	1,45	85,1	6,33	0,00	1,39	0,00	0,00	2,18	0,00	3,36	0,00	34,3	0,00	97,8	–
D2	SegNet RVB	84,9	74,7	64,1	38,0	89,9	68,4	58,4	51,4	36,7	39,8	61,0	45,7	55,2	47,9	76,9	66,9	98,7	96,7
	SegNet MSI	86,4	76,0	66,8	43,4	92,4	72,3	51,0	59,5	22,9	50,0	67,4	48,0	53,5	41,0	77,0	66,7	98,8	97,7

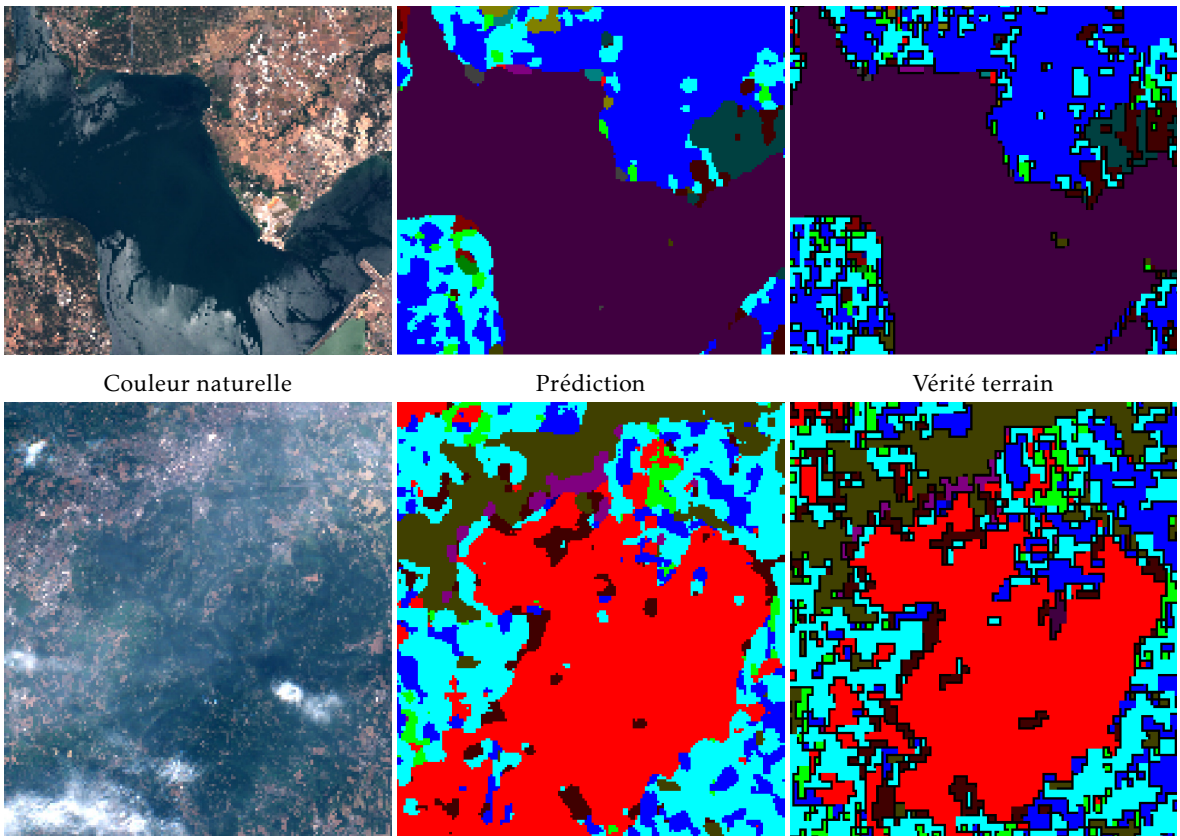


Figure 4.3 – Exemples de prédictions du modèle SegNet MSI entraîné sur D2 (avec nuages).

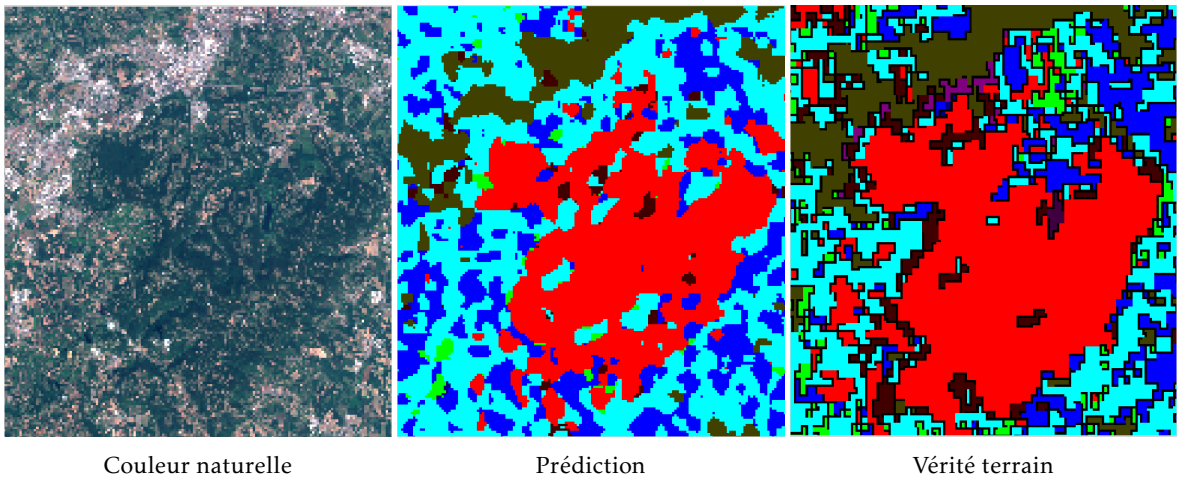


Figure 4.4 – Exemples de prédictions du modèle SegNet MSI entraîné sur D1 (sans nuage).

4.2 Imagerie hyperspectrale

Jusqu'ici, nous avons pu constater que l'interprétation d'images couleur RVB et infrarouges classiques bénéficie considérablement de l'introduction des réseaux de neurones convolutifs profonds. Qui plus est, la Section 4.1.2 nous a permis d'étendre les CNN au cas multispectral et d'exploiter à bon escient les longueurs d'onde supplémentaires afin d'identifier des objets indétectables précédemment. En réalité, chaque matériau possède une signature spectrale caractéristique définie par la façon dont il réfléchit la lumière. Il paraît donc naturel de chercher à acquérir simultanément la réponse en intensité sur l'ensemble du spectre lumineux afin de réaliser une analyse fine des surfaces [15, 19].

Cette logique est à l'origine du développement des capteurs hyperspectraux. Ceux-ci possèdent une faible résolution spatiale, mais une très grande résolution spectrale et sont capables de mesurer la réponse lumineuse d'un objet sur plusieurs centaines de bandes spectrales régulièrement espacées. Les méthodes de *deep learning* purement orientées vision ne peuvent pas se transposer directement à de telles données. En effet, la dimension spectrale prédomine devant les dimensions spatiales caractéristiques des objets d'intérêt. À titre d'exemple, une acquisition hyperspectrale aérienne typique présentera une résolution au sol au mieux de 1 m/px et une résolution spectrale d'environ 10 nm/bande, pour 200 bandes spectrales entre 0,4 µm et 2,5 µm. Si l'on considère une maison individuelle de $12\,\text{m} \times 10\,\text{m}$ [3], cet objet en hyperspectral sera décrit par un tenseur de taille $12 \times 10 \times 200$. En comparaison, les acquisitions aériennes RVB en EHR présentent une résolution spatiale de l'ordre de 10 cm/px sur 3 bandes entre 0,4 µm et 0,7 µm. La même maison serait alors caractérisée par un tenseur de dimensions $120 \times 100 \times 3$. Si le volume de données est similaire (24 000 scalaires contre 36 000), leur répartition est nettement différente. En pratique, on parle ainsi de cube hyperspectral, ou *hypercube* (cf. Figure 4.5). Enfin, compte-tenu de la faible résolution spatiale des capteurs hyperspectraux, un hypercube couvre une même zone avec moins de pixels qu'une image couleur. Le nombre d'échantillons annotés pour l'entraînement de modèles supervisés sera donc plus faible que précédemment. Ces deux propriétés sont les principaux obstacles auxquels nous allons faire face pour mettre en œuvre l'apprentissage profond sur des hypercubes.

La Section 4.2.1 rappelle les fondamentaux de l'imagerie hyperspectrale. La Section 4.2.2 détaille les jeux de données hyperspectraux mis à la disposition de la communauté scientifiques, tandis que la Section 4.2.3 réalise un bref tour d'horizon des approches de classification traditionnellement mises en œuvre sur ces images. Enfin, la Section 4.2.4 clôture cette partie avec une étude comparative, théorique et expérimentale, des modèles de réseaux profonds pour la cartographie à partir de données hyperspectrales. Le lecteur averti des spécificités liées à l'imagerie hyperspectrale pourra directement se diriger vers cette dernière section.

4.2.1 Principes physiques de l'imagerie hyperspectrale

Physiquement, un capteur hyperspectral mesure [4] l'intensité (en unité de luminance spectrale) du flux lumineux ϕ par unité de surface et par unité d'angle solide. Il s'agit d'une grandeur physique qui a pour unité le $\text{W}\cdot\text{m}^{-2}\cdot\text{sr}^{-1}$. Le capteur mesure ce flux lumineux sur un ensemble de bandes radiométriques régulièrement espacées, dont la largeur varie aux alentours de 10 nm. Pour chaque pixel, c'est-à-dire pour chaque unité de surface, le capteur échantillonne ainsi la signature spectrale de la surface sur plusieurs centaines de longueurs d'onde. Chacun de ces spectres peut se représenter sous la forme d'une courbe de réponse

3. D'après le ministère de l'environnement, la surface de plancher moyenne des maisons françaises était de $121\,\text{m}^2$ en 2015 (« *Le prix des terrains à bâtir en 2015* »).

4. Les caméras multispectrales et hyperspectrales fonctionnent généralement sur un mode *push broom* réalisant une acquisition ligne par ligne, différent des appareils photographiques habituels. Les détails techniques liés aux capteurs et à leur étalonnage ne seront pas abordés dans ce manuscrit.

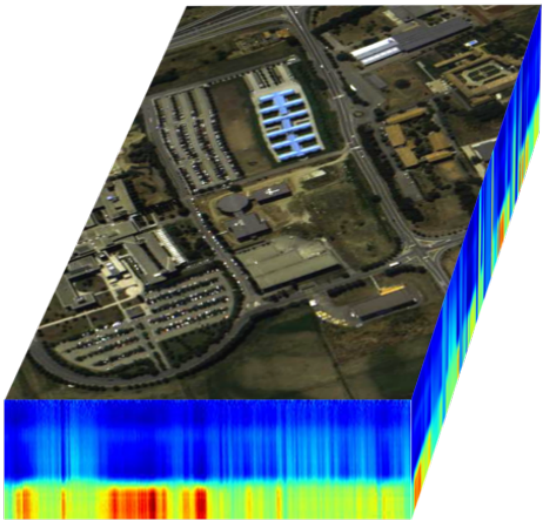


Figure 4.5 – Exemple de cube hyperspectral sur le jeu de données *Pavia University*.

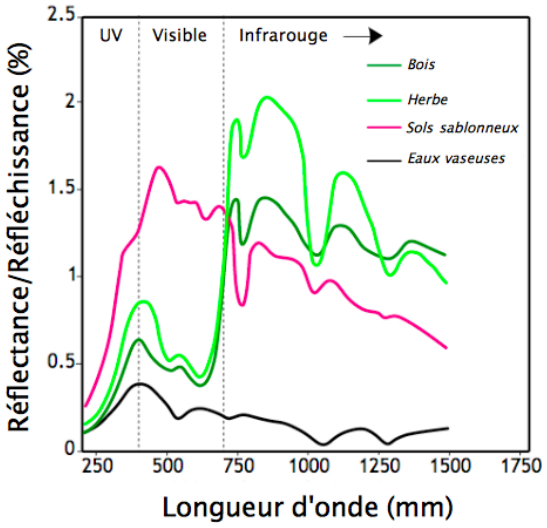


Figure 4.6 – Exemple de réflectances caractéristiques de diverses surfaces terrestres.
Crédits image : Arbeck (Wikimedia Commons, CC-BY-SA 3.0)

spectrale comme celles de la Figure 4.6. Ce flux lumineux intègre la lumière émise et réfléchie par l'objet, mais aussi celle diffusée par l'environnement, qui s'ajoute à la mesure.

En observation de la Terre, les acquisitions sont réalisées depuis le sommet de l'atmosphère (en satellite) ou depuis l'atmosphère même (en aéroporté). Or, l'atmosphère n'est pas neutre et altère le signal lumineux lors de sa propagation. Les images satellitaires sont régulièrement perturbées par les nuages, les effets de brumes et les aérosols en suspension dans l'air. Compte-tenu de la finesse des bandes d'acquisition, les capteurs hyperspectraux sont particulièrement sensibles à ces phénomènes. En ce qui nous concerne, nous nous intéressons aux surfaces et aux matériaux présents sur le sol. La mesure pertinente est donc la réflectance du sol, définie comme le rapport entre le flux émis par celui-ci et le flux incident :

$$\rho = \frac{\Phi_{réfléchi}}{\Phi_{incident}} \quad . \tag{4.1}$$

La réflectance ρ est un indicateur du pouvoir réfléchissant d'un objet pour une longueur d'onde donnée (on parle également d'albédo du matériau). C'est une grandeur sans unité comprise entre 0 (surface complètement absorbante) et 1 (surface totalement réfléchissante). En règle générale, un matériau renvoyant plus de 80% de la lumière blanche apparaît blanc, tandis qu'un matériau réfléchissant moins de 3% apparaît noir. Comme pour la luminance, on considère la réflectance radiométrique ρ_λ, qui varie en fonction de la longueur d'onde. La réflectance est préférée à la luminance car il s'agit d'une propriété intrinsèque au matériau, indépendante de l'environnement extérieur. La courbe de réflectance d'un matériau correspond de fait à une signature spectrale dotée d'un fort pouvoir discriminant (cf. Figure 4.6). Lorsque c'est possible, on cherchera donc à travailler sur des données en réflectance.

Corrections environnementales Passer de la luminance à la réflectance nécessite d'éliminer l'influence de l'environnement sur l'intensité lumineuse mesurée par le capteur. La compensation des phénomènes perturbatoires impliquent des techniques dites de correction atmosphérique [18, 48, 10]. Celles-ci permettent de réduire l'influence de l'atmosphère sur la mesure [24] et transforment les images de luminance en images de réflectance. Pour ce faire, les spécialistes conçoivent des modèles d'atmosphère qu'ils inversent afin d'estimer

puis d'éliminer l'influence de la diffusion lumineuse et des phénomènes radiatifs. Généralement, ces modèles nécessitent de connaître les paramètres environnementaux tels que l'ensoleillement. Ces informations peuvent être obtenues *a posteriori*, grâce aux éphémérides, ou *in situ* en embarquant un capteur d'ensoleillement sur le dos de l'appareil. Les capteurs hyperspectraux portatifs éclairent directement la cible et permettent ainsi de s'affranchir des conditions environnementales.

Par ailleurs, notons que le calcul de la réflectance se fait le plus souvent sous hypothèse de planarité du sol. Le relief naturel du terrain et la présence d'objets surélevés introduisent en effet réflexions et occlusions indésirables. Les premières peuvent conduire à des surilluminations lorsque plusieurs réflexions convergent vers le même point, tandis que les secondes génèrent des ombres. Certaines techniques de correction incluent donc parfois le MNE afin de prendre en compte la géométrie de la scène, notamment en milieu urbain [9].

Dans tous les cas, il est nécessaire de garder à l'esprit que toute correction, aussi maîtrisée soit-elle, est imparfaite et susceptible d'introduire des erreurs et des incertitudes dans les données.

Visualisation Contrairement à la vision humaine, la plage de fonctionnement d'une caméra hyperspectrale s'étend bien au-delà du domaine visible. La plupart des capteurs balaient les longueurs d'onde de l'ultra-violet (300 nm) jusqu'à la limite de l'infrarouge moyen (3000 nm) par tranches de 10 nm. En comparaison, le spectre visible ne couvre que les longueurs d'onde de 300 nm à $\simeq$ 700 nm. Les écrans de la vie courante utilisent le mode RVB et produisent une image comme l'agrégation de trois cartes d'intensité en rouge, vert et bleu. Cette approche correspond aux trois types de récepteurs situés dans l'œil humain. Néanmoins, une image hyperspectrale correspond à un cube de données au sein duquel chaque pixel contient une réponse spectrale complète. Ces signatures spectrales caractérisent les surfaces et les matériaux lorsqu'ils sont purs. En pratique, la faible résolution spatiale implique que le spectre mesuré pour un pixel soit un mélange de différents matériaux, d'autant plus dans le cas de la végétation.

Compte-tenu de la différence des résolutions spectrales entre l'hyperspectral (très fine devant celle des yeux humains) et de l'imagerie RVB classique, il n'y a pas d'équivalence entre les deux représentations. Une image hyperspectrale contient nettement plus d'informations que l'image RVB à résolution spatiale identique. En outre, s'il est possible de reconstruire une image RVB à partir de bandes spectrales bien choisies dans l'image hyperspectrale, les différences de résolution font qu'il ne s'agit que d'une pseudo-image qui n'aurait pas été vue de cette façon par des yeux humains. En effet, un appareil photo fonctionne en captant séparément la lumière rouge, verte et bleue grâce à un filtre, de façon à simuler le fonctionnement de l'œil humain. Un capteur hyperspectral fera généralement une acquisition ligne par ligne du spectre complet, décomposé par un prisme (capteur dit "*pushbroom*"). Les deux modes d'acquisition ne sont ainsi pas comparables.

4.2.2 Jeux de données

Plusieurs acquisitions hyperspectrales annotées ont été rendues publiques afin d'étudier les techniques de cartographie automatique sur ces données [5]. Nous présentons ici les jeux de données publics les plus utilisés par ordre de popularité.

Comme nous allons le voir, une des difficultés majeures en apprentissage statistique pour le traitement d'images hyperspectrales est le faible nombre d'échantillons disponibles. Compte-tenu des différences de capteurs, de conditions d'exposition et d'étalonnage, il est difficile d'exploiter conjointement plusieurs acquisitions. Or, prises individuellement, les images hyperspectrales annotées offertes à la communauté scientifique sont de taille très faible en comparaison des banques d'images RVB habituelles. Cela complique l'évaluation

5. http://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes

des méthodes d'apprentissage supervisées. Il existe bien une base de données à grande échelle d'acquisitions hyperspectrales [6], réalisées avec le capteur *Airborne Visible/Infrared Imaging Spectrometer* (AVIRIS) sur le territoire américain, mais ces images ne sont pas annotées.

Pavia Pavia est un jeu de données acquis via le capteur ROSIS avec une résolution au sol de 1,3 m sur la ville de Pavie, en Italie. Il est divisé en deux scènes : Pavia University (103 bandes, 610 px × 340 px) et Pavia Centre (102 bandes, 1096 px × 715 px). 9 classes d'intérêt sont annotées, couvrant différents matériaux urbains (brique, asphalte, métaux), l'eau et la végétation sur 50% de l'image.

Il s'agit d'un des principaux jeux de données de référence dans la communauté, notamment car les deux scènes se trouvent parmi les plus grandes images hyperspectrales annotées disponibles. En outre, l'utilisation du même capteur sur les deux images permet de tester des méthodes de transfert de connaissances en hyperspectral.

Indian Pines Indian Pines est un jeu de données acquis en utilisant le capteur américain AVIRIS. La scène couvre une surface agricole sur 224 bandes spectrales pour 145 px × 145 px, avec une résolution au sol de 3,7 m/px. La majorité de l'image consiste en des champs d'une dizaine de cultures différentes, le reste étant occupé par de la végétation dense. 16 classes sont annotées, dont certaines très rares (moins de 100 échantillons). Les bandes d'absorption de l'eau (108→112, 154→167 et 224) sont généralement enlevées.

En dépit de sa faible taille, il s'agit d'un des principaux jeux de données de référence dans la communauté. Les classes les plus rares ne sont parfois pas prises en compte pour évaluer les algorithmes de classification.

Salinas Salinas est un jeu de données utilisant également le capteur AVIRIS. La scène comporte 512 × 217 échantillons à 3,7 m/px. Les bandes d'absorption de l'eau (108→112, 154→167 et 224) sont généralement enlevées. 16 classes sont annotées, majoritairement concernant les différentes cultures observées, la végétation et le type de sol.

Kennedy Space Center (KSC) Le jeu de données KSC utilise le capteur AVIRIS avec une résolution au sol de 18 m/px. Il s'agit d'une acquisition réalisée sur le centre spatial Kennedy à Cape Canaveral (Floride, États-Unis). L'ensemble de l'image est de dimensions 512×614. Les bandes d'absorption de l'eau et celles avec un rapport signal/bruit faibles sont retirées pour ne conserver que les 176 bandes les plus informatives. 13 classes concernant les différents types de végétation occupant les abords du centre ont été annotées.

Botswana Botswana est un jeu de données acquis sur le delta du Okavango à l'aide du capteur Hyperion embarqué par le satellite EO-1 de la NASA, à une résolution de 30 m/px sur 242 bandes. L'image complète est de dimensions spatiales 1476 × 256. Seules les 145 bandes 10→55, 82→97, 102→119, 134→164 et 187→220 sont conservées, les autres correspondant aux bandes d'absorption de l'eau et à des bandes mal calibrées. 14 classes d'intérêt sont annotées concernant différents types de végétation et de marécages.

DFC 2018 Le jeu de données DFC 2018 correspond à une acquisition de dimensions 2384 × 1202 sur le centre-ville de Houston, Texas (États-Unis) à l'aide d'une caméra hyperspectrale aérienne. L'acquisition couvre le domaine 380–1050 nm à l'aide de 48 bandes à une résolution de 1 m/px. 20 classes d'intérêt sont définies, incluant des objets urbains (bâtiments, routes de différents types, rails, voitures, trains…) mais aussi la végétation (saine, stressée, décidue, sempervirent). Ce jeu de données est issu de la compétition *Data Fusion Contest* 2018, détaillé dans l'Annexe A.1.3. La moitié des annotations sont publiques, tandis que l'autre est réservée par les organisateurs pour une évaluation indépendante sur un serveur en ligne.

6. *AVIRIS Data Portal* : https://aviris.jpl.nasa.gov/alt_locator/

Récapitulatif Les caractéristiques des différents jeux de données publics identifiés sont listées dans le Tableau 4.5. Le principal élément qui en ressort est le faible nombre d'échantillons annotés disponibles sur chacun des jeux de données. Le capteur AVIRIS est utilisé sur plusieurs scènes, mais les classes identifiées ne sont pas cohérentes d'une acquisition à l'autre, limitant le potentiel de réutilisation des modèles.

Tableau 4.5 – Récapitulatif des principaux jeux de données publics annotés en imagerie hyperspectrale.

Jeu de données	Pixels	Bandes	Domaine	Résolution	Annotations	Classes	Mode
Pavia	991 040	103	0,43–0,85 µm	1,3 m/px	50 232	9	Aérien
Indian Pines	21 025	224	0,4–2,5 µm	3,7 m/px	10 249	16	Aérien
Salinas	111 104	224	0,4–2,5 µm	3,7 m/px	54 129	16	Aérien
KSC	314 368	176	0,4–2,5 µm	18 m/px	5211	13	Aérien
Botswana	377 856	145	0,4–2,5 µm	30 m/px	3248	14	Satellite
DFC 2018	5 014 744	48	0,38–1,05 µm	1 m/px	547 807	20	Aérien

4.2.3 Approches traditionnelles

Nous présentons dans cette section un bref tour d'horizon des techniques courantes mises en œuvre dans l'état de l'art pour le traitement d'images hyperspectrales, avec une attention particulière accordée aux méthodes supervisées.

prétraitements et normalisations

Comme nous l'avons vu, les données hyperspectrales brutes sont rarement exploitables directement. En plus des corrections atmosphériques et de l'orthorectification afin d'obtenir des cartes de réflectance géo-référencées, il est courant d'effectuer diverses opérations de normalisation des données.

En premier lieu, il est fréquent de retirer certaines longueurs d'onde difficiles à exploiter. Selon la calibration, certaines bandes peuvent être saturées et écrasent la dynamique des spectres. À l'inverse, l'humidité atmosphérique dégrade le signal dans les bandes d'absorption de l'eau. Dans l'ensemble, seules les bandes avec un rapport signal sur bruit acceptable sont conservées, ce qui permet de réduire la dimensionalité des données avec une perte minimale d'information.

Ensuite, il est souvent préférable de normaliser statistiquement les spectres. Plusieurs approches peuvent être utilisées selon les propriétés devant être mises en valeur :

— Si les formes des spectres sont plus importantes que leurs amplitudes, on utilisera l'angle spectral, version normalisée du spectre :

$$X^* := \frac{X}{\|X\|} \ ,$$

— La normalisation des moments statistiques de premier et second ordres (moyenne nulle et variance unitaire), globale ou pour chaque bande, permet de faire apparaître et de retirer des anomalies (à $\pm 5\sigma$, par exemple) :

$$I^* := \frac{I - m_I}{\sigma_I^2} \text{ avec } m_I \text{ la moyenne de I et } \sigma_I \text{ son écart-type,}$$

— Enfin, la normalisation globale dans $[0,1]$ est couramment utilisée pour simplifier les manipulations numériques. Alternativement, elle permet de donner la même importance à toutes les longueurs d'onde lorsqu'elle est appliquée bande par bande :

$$I^* := \frac{I - \min(I)}{\max(I) - \min(I)} \ .$$

Les valeurs aberrantes, par exemple supérieures au 98e percentile ou au-delà de $m_I + 5\sqrt{\sigma_I}$, peuvent être tronquées ou supprimées pour limiter leur influence. Cela permet de prendre en compte des anomalies dues aux erreurs de correction, à des matériaux particulièrement réfléchissants (comme les métaux) ou aux réflexions multiples.

Soulignons que, dans un cadre parfait, un pixel d'une image hyperspectrale correspond à la réflectance du matériau observé sur une unité de surface. Toutefois, la résolution spatiale des images fait qu'un pixel correspond à une surface couvrant plusieurs matériaux, produisant ainsi des spectres de mélange. Concrètement, si $\varphi_1, \ldots, \varphi_n$ désignent les spectres purs de l'ensemble des matériaux de la scène, alors en un pixel (i, j), le spectre local observé $\phi_{i,j}$ sera une fonction f des φ_i :

$$\phi_{i,j} \quad = \quad f_{i,j}(\varphi_1, \ldots, \varphi_n) \quad \simeq \quad \sum_{k=1}^{n} \lambda_k \varphi_k \ . \tag{4.2}$$

Dans le cas où la surface est plane, on peut faire l'hypothèse que f est une simple combinaison linéaire, où le coefficient de pondération λ_k correspond à la proportion du matériau k dans la surface observée [7].

Un certain nombre de travaux s'intéressent à l'inversion de ce problème sous la dénomination « démélange » [44]. La classification la plus simple consiste ainsi à déterminer les matériaux purs qui composent la scène et de chercher à calculer des cartes d'abondance. Les spectres de référence des matériaux purs sont appelés les *endmembers* [8] et constituent une base de décomposition des spectres mélangés. Les cartes d'abondance correspondent alors aux proportions des différents matériaux en chaque point. Généralement, en connaissant les spectres purs S_k et l'image I, il est possible d'inverser le système linéaire pour obtenir les coefficients λ_k du mélange en chaque point. Ces méthodes reposent principalement sur des mécaniques d'algèbre linéaire et des méthodes numériques d'inversion de problème. Des méthodes d'apprentissage, par exemple par *clustering*, permettent d'obtenir les *endmembers* quand ils sont inconnus. L'identification des *endmembers* et le démélange est hors du cadre de travail considéré pour nos travaux.

Classification de spectres

Les approches de classification de données hyperspectrales les plus simples opèrent pixel à pixel et traitent ainsi les spectres indépendamment les uns des autres. Nous présentons ici quelques-unes de ces approches unidimensionnelles. Nous écartons volontairement les approches expertes pour ne considérer que les méthodes d'apprentissage statistique.

Une première approche consiste à réduire la dimension spectrale des données afin de lutter contre la malédiction de la dimensionalité. En effet, compte-tenu de la haute résolution spectrale des imageurs, les réflectances voisines tendent à être fortement correlées, et la signature spectrale contient une information très redondante. Il est donc souvent intéressant de réduire la taille des données en ne considérant que les bandes contenant de l'information discriminante [29, 7]. Rodarmel et Shan [50] applique ainsi une analyse en composantes principales (ACP) aux spectres avant leur classification. Le calcul des indices physiques comme le NDVI ou NDWI rentre également dans ce type d'approches.

La classification se fait ensuite de façon traditionnelle, en utilisant des modèles statistiques classiques : arbres de décision et forêts aléatoires, machines à vecteurs de support (SVM), etc. La réduction de dimension permet de simplifier l'espace de représentation et facilite donc l'apprentissage.

Cependant, l'approche purement spectrale n'est pas satisfaisante dans la mesure où elle n'exploite pas la structure spatiale des objets. En effet, les progrès technologiques permettent

7. Certains mélanges de matériaux présentent également des propriétés non-linéaires, mais on traite alors ces cas à part.

8. En minéralogie, un *endmember* est un minéral en bout de chaîne de pureté. La plupart des minéraux sont des solutions solides, c'est-à-dire des mélanges de ces *endmembers*).

d'améliorer la résolution des capteurs et donc d'augmenter le nombre de pixels acquis pour une même surface. Des pixels voisins partageront vraisemblablement de nombreuses propriétés spectrales, et des structures peuvent alors apparaître (par exemple, les bâtiments ont généralement des formes polygonales tandis que la végétation présente une apparence fractale). Prendre en compte l'aspect spatial permet de rendre le modèle plus robuste en apprenant ces dépendances structurelles. Il existe trois grandes familles d'approches selon l'importance donnée à l'aspect spatial dans le processus de classification.

L'approche la plus simple consiste à effectuer une classification spectre par spectre en utilisant un modèle unidimensionnel, puis à régulariser *a posteriori* les prédictions. Les modèles graphiques comme les CRF se prêtent particulièrement bien à cette application [62]. La régularisation spatiale intervient ainsi durant une seconde étape de traitement, indépendante de la première. À l'inverse, il est possible de faire intervenir l'aspect spatial en amont en utilisant le principe de classification par région, déjà présenté dans la Section 3.1. Tarabalka, Chanussot et Benediktsson [57] et Fauvel et al. [21] décrivent plusieurs méthodes en deux étapes : premièrement une segmentation de l'image hyperspectrale, puis des prédictions pixelliques agrégées par région afin d'introduire une cohérence spatiale locale. Enfin, il existe également des stratégies s'appuyant sur des caractéristiques spatiales-spectrales. C'est l'approche originellement poursuivie afin d'exploiter la corrélation entre pixels spatialement proches pour le calcul des *endmembers* [46, 17] en utilisant un mélange de classifieurs spatial et spectral. Les modèles plus récents utilisent des noyaux spécifiquement conçus pour travailler sur des voisinages locaux de spectres, de taille fixe ou adaptative, pour en extraire une combinaison de caractéristiques spatiales et spectrales. Notamment, Camps-Valls et al. [8] ont introduit la possibilité de travailler sur des SVM à noyau spatial-spectral pour les données hyperspectrales, technique qui sera ensuite largement réutilisée dans la littérature [56, 20]. Plus récemment, Cui, Chapel et Lefèvre [16] ont introduit des SVM à noyaux adaptés aux profils d'attributs morphologiques, tandis que Tuia, Flamary et Courty [59] ont développé une méthode adaptative de sélection de noyaux de convolutions à partir de filtres aléatoires dont le principe s'approche de l'apprentissage automatique de représentations.

4.2.4 Apprentissage profond et imagerie hyperspectrale

Les approches présentées jusqu'ici utilisent des modèles statistiques superficiels, sans apprentissage de représentation. Néanmoins, à partir de 2013, la communauté en classification d'images hyperspectrales a commencé à mettre en œuvre des réseaux de neurones profonds spécifiquement adaptés aux hypercubes.

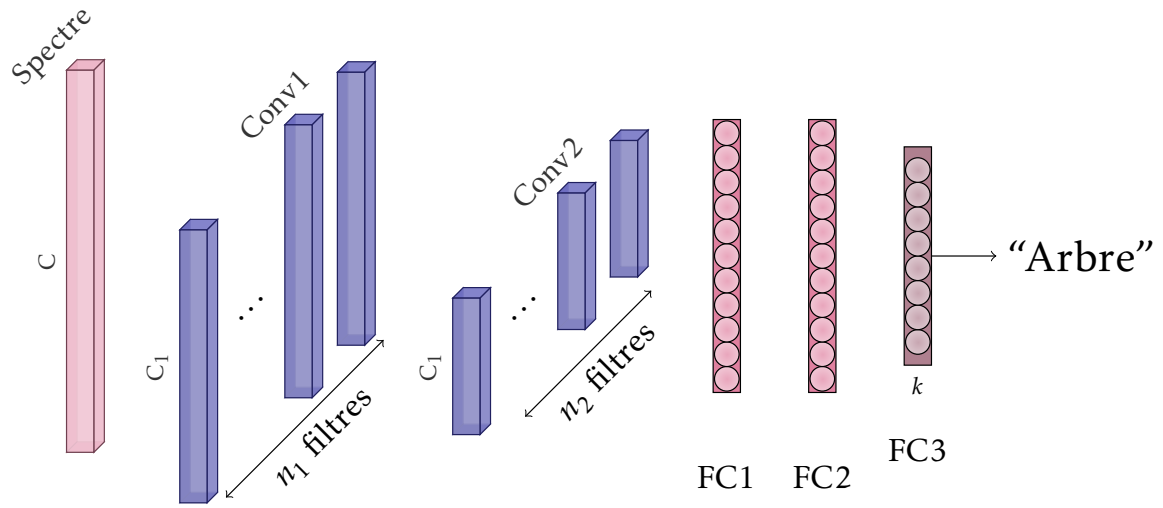


Figure 4.7 – CNN unidimensionnel pour la classification de spectres de Hu et al. [28].

L'évolution la plus simple consiste à remplacer les classifieurs standard (SVM ou forêts aléatoires) par un perceptron multicouche. Conceptuellement, le processus est identique mais si le réseau est assez profond, il peut s'avérer plus expressif et doté d'une meilleure capacité de discrimination. Cette approche existe en réalité depuis les années 2000 [25, 49] en utilisant des réseaux à une ou deux couches cachées. Elle a été réactualisée par Hu et al. [28] en 2015 en utilisant des CNN unidimensionnels appliqués sur les spectres individuels (cf. Figure 4.7). Une alternative originale consiste à traiter les spectres comme des séquences desquelles un *Recurrent Neural Network* (RNN) peut extraire des motifs. Mou, Ghamisi et Zhu [41] ont ainsi appliqué avec succès des architectures de RNN, initialement conçues pour les séries temporelles, aux données hyperspectrales.

Dès lors que les réseaux profonds entrent en jeu, peu d'articles s'embarrassent des problématiques de sélections de bandes, de rejet des valeurs saturées ou d'analyse des phénomènes physiques mis en jeu. En effet, les modèles profonds excellant pour l'apprentissage de représentations, il devient alors possible de traiter directement les données brutes normalisées, y compris les bandes spectrales bruitées ou saturées. En pratique, la robustesse des réseaux profonds permet de ne pas avoir à se préoccuper de ces considérations, le modèle éliminant naturellement les données non informatives.

Les autoencodeurs ont largement contribué à cette tendance. En effet, les capacités de compression du signal des autoencodeurs ont permis d'entraîner des modèles de réduction de dimension avec une perte d'information minimale, générant des représentations nettement plus discriminantes qu'une ACP, avec de nombreuses applications pratiques en débruitage [63]. Les représentations ainsi apprises peuvent être finalement utilisées pour la classification par n'importe quel modèle statistique [23].

Comme précédemment, il existe des approches spatiales-spectrales basées sur des descripteurs combinant une caractéristique spectrale, dérivée de la réponse radiométrique, et une caractéristique spatiale, dépendant des pixels voisins. Un descripteur classique consiste à concaténer le spectre du pixel considéré avec le résultat d'une ACP appliquée sur un voisinage local de dimensions $w \times h$ dont on conserve les K premières composantes (généralement, $w = h \simeq 8$ et $K = 3$). Ce vecteur est ensuite utilisé pour entraîner un classifieur profond, supervisé ou non : DBN [32, 11], RBM [34, 40] ou cascade d'autoencodeurs [12, 37, 55, 61].

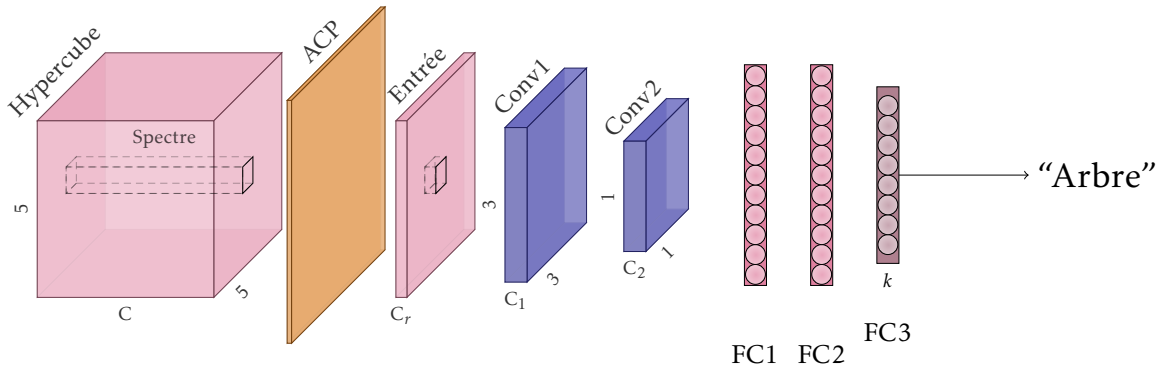


Figure 4.8 – Architecture hybride ACP+CNN de Makantasis et al. [39] pour la classification d'hypercubes.

Toutefois, le retour sur le devant de la scène des CNN au début des années 2010 a également influencé la communauté de l'imagerie hyperspectrale. Ces réseaux sont initialement prévus pour traiter des images RVB ou en niveaux de gris et manipulent donc des filtres convolutifs 2D. Makantasis et al. [39] et Slavkovikj et al. [53] utilisent une architecture hybride alternant convolutions spatiales et réductions de dimension (par ACP chez Makantasis et al. [39] et sous-échantillonnage chez Slavkovikj et al. [53]). Les caractéristiques résultantes sont ensuite vectorisées et transmises à un perceptron multicouche réalisant la classification, comme schématisé par la Figure 4.8. L'intérêt de cette approche est de pouvoir apprendre

automatiquement la représentation des données adaptée à la classification. Zhao et al. [68] étendent cette technique au cadre semi-supervisé en utilisant des autoencodeurs convolutifs multiéchelles. Dans le cadre non-supervisé, Romero, Gatta et Camps-Valls [51] proposent un CNN pour l'extraction de caractéristiques effectuant une réduction de dimension en prenant en compte le spectre, mais aussi ses voisins, afin d'obtenir une représentation parcimonieuse des spectres. Enfin, Zhao et Du [67] et Yue et al. [66] proposent une approche mixte utilisant un CNN 2D comme extracteur de caractéristiques spatiales, qu'ils combinent à un CNN 1D pour générer un descripteur spatial-spectral.

Bien que performantes, ces architectures différencient dans leur traitement les aspects spatiaux et spectraux de l'hypercube. Pourtant, les approches traditionnelles ont montré la pertinence des noyaux spatiaux-spectraux et de l'apprentissage conjoint qu'ils permettent. Plusieurs travaux ont donc introduit des convolutions tridimensionnelles permettant d'apprendre des noyaux opérant directement sur le cube de données. En particulier Ben Hamida et al. [5] et Chen et al. [13] proposent des CNN alternant convolutions 3D et convolutions 1D pour la réduction de dimension. Luo et al. [36] suggère une approche alternative en remplaçant l'ACP de Makantasis et al. [39] par une couche convolutive 3D réalisant la réduction de dimension, suivie d'un CNN 2D traditionnel. Comme d'habitude, la classification s'opère pixel à pixel par deux couches entièrement connectées en fin de réseau. Lee et Kwoon [31] étendent cette structure en un FCN dont la première couche extrait une caractéristique spatiale-spectrale en utilisant deux convolutions parallèles, une en 1D et une en 3D, en s'inspirant du module *Inception*. Le reste de l'architecture présente une structure résiduelle enchaînant les convolutions 1D.

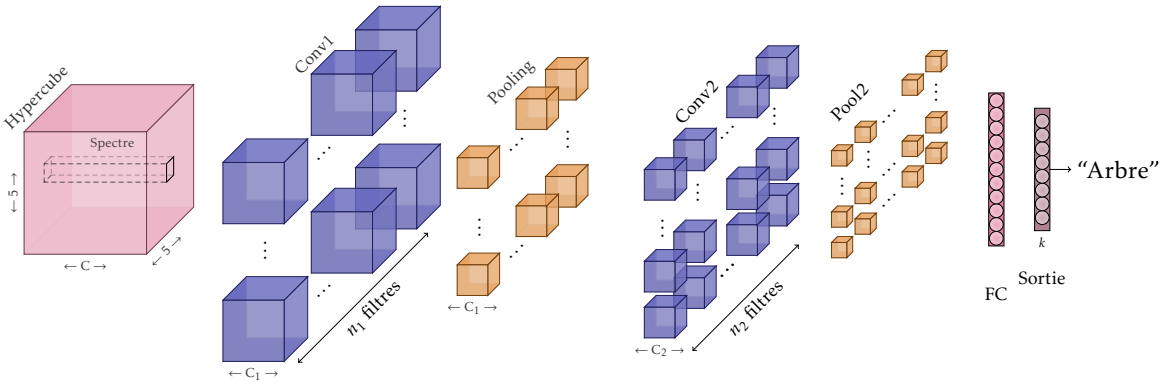


Figure 4.9 – CNN 3D de Chen et al. [13] pour la classification d'hypercubes.

Enfin, des réseaux convolutifs entièrement 3D ont finalement été introduits, adaptant l'architecture canonique des CNN pour l'imagerie RVB aux hypercubes. Diverses architectures ont été proposées [33], incluant des variantes bien connues de la communauté vision par ordinateur, comme l'extraction de caractéristiques multiéchelles [27] et l'entraînement semi-supervisé [35]. Dans l'ensemble, il s'agit d'une extension du modèle canonique des CNN de LeCun et al. [30] aux données tridimensionnelles, comme schématisé dans la Figure 4.9.

Malgré la profusion de littérature concernant la classification d'images hyperspectrales, il n'existe pas de cadre standardisé d'évaluation des modèles. En particulier, chaque auteur considère un ou plusieurs des jeux de données présentés en Section 4.2.2 avec des partitions entraînement/validation/test différent. En outre, les implémentations mises à disposition de la communauté scientifique sont particulièrement rares en comparaison des pratiques habituelles en vision par ordinateur. Par conséquent, nous développons une boîte à outils modulaire d'apprentissage profond pour la cartographie automatisée à partir d'images hyperspectrales appelée *DeepHyperX* [9]. Celle-ci comporte nativement plusieurs approches supervisées, allant de la SVM linéaire aux CNN 3D de l'état de l'art en passant par les

9. https://gitlab.inria.fr/naudeber/DeepHyperX

CNN 1D. Ces modèles peuvent être entraînés et validés sur divers jeux de données de la littérature comme Pavia Center et Pavia University, Indian Pines, Kennedy Space Center ou *Data Fusion Contest* (DFC) 2018. Les hyperparamètres les plus courants peuvent être ajustés afin d'évaluer l'influence des dimensions du voisinage local considéré, du nombre d'échantillons d'apprentissage ou de la stratégie d'optimisation. Ce logiciel nous permet de comparer les performances des modèles de l'état de l'art dans un cadre unifié.

Techniquement, cette boîte à outils est écrite en Python [22] et consiste en une interface construite autour des bibliothèques PyTorch [47] et scikit-learn [45]. Les réseaux de neurones profonds sont implémentés à l'aide de PyTorch, afin de permettre une exécution sur *Central Processing Unit* (CPU) comme sur GPU, tandis que les SVM utilisent scikit-learn. Plusieurs jeux de données publics sont préconfigurés afin de faciliter l'expérimentation. L'architecture modulaire de la boîte à outils permet aux programmeurs de facilement ajouter de nouveaux jeux de données ou de nouveaux modèles de réseaux profonds pour tester de nouvelles idées ou évaluer les performances sur des acquisitions privées.

Dans la suite, nous évaluons plusieurs architectures profondes de la littérature pour la classification d'images hyperspectrales de télédétection. À notre connaissance, il s'agit de la première étude systématique des différents modèles de réseaux convolutifs introduits dans l'état de l'art. La majorité des articles réalise des expériences légèrement différentes les unes des autres, soit en excluant certaines classes, soit en considérant des partitions entraînement/validation différentes. De plus, l'approche la plus fréquente consiste à sélectionner un des jeux de données publics, à entraîner un modèle sur un ensemble de pixels tirés aléatoirement dans l'image et à valider ses performances sur le reste de l'image. Or, cette approche n'est pas réaliste, dans la mesure où des pixels proches les uns des autres seront fortement corrélés. Par conséquent, le jeu de validation sera très proche du jeu d'apprentissage et les métriques de classification ne seront pas réellement indicatives de la capacité de généralisation du modèle. Au contraire, ces pratiques récompensent le sur-apprentissage et sont généralement découragées par la communauté de l'apprentissage automatique. Nous suivons donc les approches classiques d'évaluation de modèles statistiques en considérant des partitions entraînement/validation *spatialement disjointes* et en moyennant les résultats sur plusieurs entraînements, si possible par validation croisée sur plusieurs partitions. Dans le cas des CNN 2D et 3D, cela garantit notamment qu'aucun pixel du jeu de validation n'aura été vu accidentellement durant l'apprentissage.

Pour la suite, nous considérons les partitions définies par l'*Institute of Electrical and Electronics Engineers* (IEEE) *Geoscience & Remote Sensing Society* (GRSS) sur leur plate-forme d'évaluation DASE [10] pour les jeux de données Indian Pines, Pavia University et DFC 2018. Les hyperparamètres sont choisis en considérant un jeu de validation séparé incluant 5% du jeu d'apprentissage.

Nous utilisons notre boîte à outils afin de comparer plusieurs réimplémentations de modèles de l'état de l'art. Ceux-ci ont été reproduits le plus fidèlement possible. Les changements appliqués sont listés ci-dessous :

- CNN 1D de Hu et al. [28]. L'algorithme d'optimisation n'étant pas spécifié dans l'article original, nous utilisons la descente de gradient stochastique avec moment.
- RNN 1D de Mou, Ghamisi et Zhu [41]. Nous utilisons la fonction d'activation *tanh* usuelle en lieu et la place de la version paramétrisée des auteurs.
- CNN 3D+1D de Ben Hamida et al. [5]. Pas de modification.
- CNN 3D de Li, Zhang et Shen [33]. Nous avons augmenté le nombre de filtres de 16 à 32 dans les couches convolutives pour une meilleure convergence.

Les CNN 3D sont entraînés sur des voisinages de dimensions 5×5. Afin d'établir des scores de référence, nous utilisons deux modèles simples : une SVM aux hyperparamètres obtenus par *grid search* et un réseau entièrement connecté 1D à trois couches utilisant ReLU [43] comme activation et auquel *Dropout* [54] est appliqué. Le déséquilibre entre les

10. *GRSS Data and Algorithm Standard Evaluation website* : http://dase.ticinumaerospace.com/

Tableau 4.6 – Résultats de classification de différents modèles de notre boîte à outils *DeepHyperX* sur les jeux de données Indian Pines, Pavia University et DFC 2018. Les meilleurs résultats sont en **gras** et les suivants sont en *italique*.

Modèle	Indian Pines		Pavia University		DFC 2018	
	Exactitude	κ	Exactitude	κ	Exactitude	κ
SVM	81,43	0,788	69,56	0,592	42,51	0,39
1D NN	**83,13**± 0,84	**0,807**± 0,009	76,9± 0,86	0,711± 0,010	41,08	0,37
1D CNN [28]	*82,99*± 0,93	*0,806*± 0,011	81,18± 1,96	0,759± 0,023	*47,01*	*0,44*
RNN [41]	79,70± 0,91	0,769± 0,011	67,71± 1,25	0,599± 0,014	41,53	0,38
3D+1D CNN [5]	74,31± 0,73	0,707± 0,008	*83,80*± 1,29	*0,792*± 0,016	46,28	0,43
3D CNN [33]	75,47± 0,85	0,719± 0,010	**84,32**± 0,72	**0,799**± 0,009	**49,26**	**0,46**

classes est géré au niveau de la fonction de coût en utilisant un rééquilibrage par fréquence médiane inverse. Les données sont augmentées par symétries verticale et horizontale. Les résultats sont détaillés dans le Tableau 4.6, incluant l'exactitude globale et le κ de Cohen sur les trois jeux de données. Les expériences ont été répétées 5 fois sur Pavia University et Indian Pines, mais seulement une fois sur DFC 2018 compte-tenu de sa taille plus importante.

Comme il était possible de s'y attendre, nous obtenons des résultats significativement inférieurs à ceux indiqués dans les articles originaux, notamment car nous utilisons une partition apprentissage/validation disjointe. Cela nous permet notamment de mettre en évidence un comportement particulier d'Indian Pines par rapport aux autres jeux de données. En effet, la prise en compte du contexte spatial sur ce jeu de données diminue en pratique les performances. Nous émettons l'hypothèse que la faible résolution spatiale d'Indian Pines (20 m/px) implique que chaque pixel est déjà un mélange de la réflectance des cultures sur 400 m^2, et que les pixels voisins n'apportent pas plus d'information discriminante. Pour Pavia University et DFC 2018, sur lesquels la résolution est plus haute, les CNN 3D sont significativement plus performants que les modèles 1D, augmentant l'exactitude globale de respectivement 3% et 2%. En particulier, le jeu de données DFC 2018 est difficile compte-tenu du grand nombre de classes similaires. Dans nos expériences, le réseau entièrement connecté 1D souffre d'un important surapprentissage et s'avère moins discriminant qu'une simple SVM. Ce surapprentissage est d'autant plus important sur le DFC 2018 car le jeu d'apprentissage est entièrement disjoint de l'image initiale, contrairement à Indian Pines et Pavia University. Enfin, le CNN 3D de Ben Hamida et al. [5] ne parvient pas à extraire de l'information spatiale discriminante sur le DFC 2018. En pratique, les deux premières couches 3D comportent trop peu de paramètres et le champ réceptif du réseau est trop faible pour correctement modéliser les relations spatiales entre pixels sur un jeu de données haute résolution.

Un obstacle majeur identifié dans ces travaux est la difficulté d'entraîner des modèles ne souffrant pas de surapprentissage. En effet, le nombre d'échantillons considéré étant faible, les modèles utilisés comportent souvent suffisamment de poids pour mémoriser le jeu de données. Augmenter le nombre d'échantillons d'apprentissage n'est pas chose facile. Notamment, les différences de capteurs rendent complexe l'apprentissage par transfert. Si des approches d'adaptation de domaine existent pour appliquer un classifieur à de nouvelles acquisitions [60], elles ne résolvent pas le problème de l'entraînement initial de celui-ci. Un palliatif possible consiste à générer des données synthétiques suffisamment réalistes pour améliorer la généralisation du modèle. Cette technique sera étudiée dans le Chapitre 6.

À noter également que la plupart des architectures actuelles se content d'effectuer une classification pixel à pixel. Comme nous l'avons vu dans le Chapitre 3, à mesure que les images hyperspectrales augmentent en résolution et en dimensions, il sera vraisemblablement nécessaire de concevoir des architectures 3D entièrement convolutives.

Enfin, il est souhaitable de voir apparaître de nouveaux jeux de données annotés en hy-

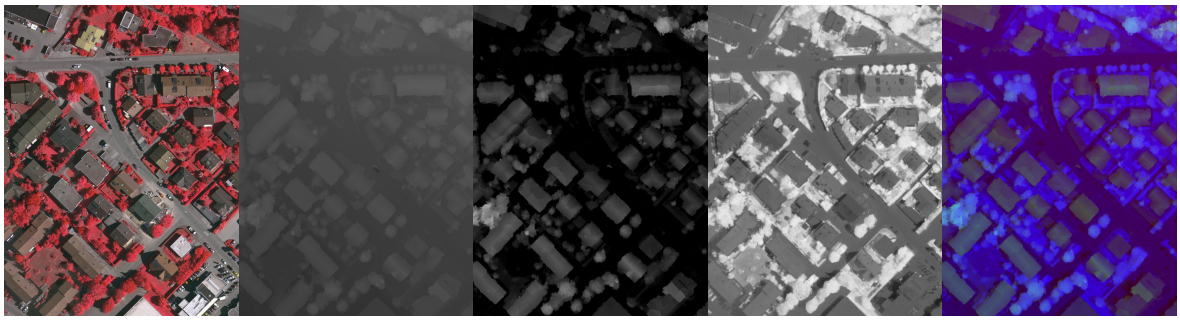
perspectral plus complexes et plus grands que ceux existants. En effet, ces derniers semblent avoir largement atteint leurs limites. L'état de l'art rapporte généralement des améliorations quantitativement incrémentales dont la pertinence statistique est discutable. Comment évaluer un modèle obtenant 99,5% de précision par rapport à un autre à 99,8% ? Un jeu de données proposant un cadre expérimental unifié permettrait de comparer équitablement diverses approches sur des tâches complexes qui ne seraient pas abordables sans imagerie hyperspectrale. Cette approche semble d'ores et déjà engagée par l'IEEE GRSS, notamment dans le cadre du DFC 2018.

Dans l'ensemble, nous avons pu mettre en évidence la difficulté pratique de mettre en œuvre des réseaux profonds pour l'apprentissage sur des données hyperspectrales. En effet, si les approches spatiales-spectrales, notamment les CNN 3D, semblent supérieures aux techniques traditionnelles de classification, nous avons pu voir que les performances réelles de ces modèles diminuent significativement lorsqu'une évaluation stricte leur est appliquée. Ainsi, nous avons développé et mis à disposition de la communauté une boîte à outils logicielle *DeepHyperX* d'apprentissage profond pour l'hyperspectral permettant de facilement comparer différents modèles sur des jeux de données communs avec un protocole standardisé. Cela permet aux thématiciens d'utiliser simplement des modèles de réseaux profonds de l'état de l'art pour la classification automatique, mais aussi aux spécialistes de l'apprentissage de valider de nouveaux modèles. Nous espérons que cet outil permettra de consolider les progrès en classification de données hyperspectrales par apprentissage profond.

4.3 Imagerie laser et modèles de terrain

Hors du domaine optique, un des capteurs de prédilection pour l'observation de la Terre est le Lidar. Il s'agit d'un capteur laser permettant d'évaluer, entre autres, la hauteur des points de la surface du globe. Lors d'une acquisition optique aérienne, il est courant d'embarquer également dans l'appareil un Lidar afin d'étudier la topologie du terrain. Nous nous intéressons dans cette section aux possibilités d'exploiter directement ces données à l'aide de FCN et nous comparons les résultats obtenus aux modèles optiques du Chapitre 3.

La Section 4.3.1 détaille dans un premier temps une approche de cartographie utilisant les modèles numériques de terrain comme seule source d'information, tandis que la Section 4.3.2 étudie la création de fausses images couleur composites agrégeant NDVI et cartes d'élévation.



Orthoimage IRRV | MNE | MNH | NDVI | Composite

Figure 4.10 – Tuile 30 du jeu de données ISPRS Vaihingen selon plusieurs modalités.

4.3.1 Modèle de terrain

Les acquisitions de nuage de points par imagerie Lidar permettent d'obtenir des modèles numériques détaillant la topologie du terrain observé : MNT, MNE et MNH. Ces points ne sont pas uniformément répartis et forment ainsi un nuage de densité variable qui ne

correspond généralement pas à une grille bien définie. Si l'obtention de ces modèles par rasterisation des nuages de points Lidar [14] ou mise en correspondance d'une paire d'images stéréo [58] est hors du cadre de cette thèse, il s'agit néanmoins d'une source de données particulièrement intéressante. En effet, les modèles de terrain permettent d'accéder à une information concernant l'élévation locale du terrain (MNT) et des objets qui s'y trouvent (MNH). La majorité des images de télédétection aéroportées et satellitaires étant acquises au nadir, celles-ci n'expriment donc pas d'information de hauteur ou de distance, contrairement aux images multimédia qui bénéficient des effets de perspective. Il en découle une absence d'occlusion et l'existence d'un facteur d'échelle unique, mais cela rend également plus complexe l'estimation de la hauteur des objets observés à partir des images optiques seules. Si les ombres portées peuvent donner une information indirecte concernant l'élévation des objets, celle-ci est toutefois peu fiable car dépendante de la topologie du terrain et surtout des conditions environnementales d'illumination (azimuth de l'acquisition, position du soleil, météo).

Or, l'environnement urbain présente de nombreux éléments surélevés pouvant se confondre avec le sol : ponts, parkings aériens, toits bétonnés ou végétalisés, végétation arborescente dense… Une analyse visuelle des cartes sémantiques obtenues par les réseaux profonds présentés dans le Chapitre 3 indique que ce sont ces éléments qui sont généralement incorrectement prédits. Les modèles de terrain permettent donc d'accéder à une information physique complémentaire à celle des images optiques pouvant ainsi renforcer l'exactitude des modèles appris.

Un modèle numérique de terrain se présente comme une image associant à chaque pixel, c'est-à-dire à chaque point associé à des coordonnées géographiques, un scalaire indiquant son élévation. Le référentiel de cette élévation peut varier et n'est pas forcément constant, dans le cas du MNH notamment. En normalisant ces données, il est possible de considérer les modèles numériques de terrain comme des images en niveaux de gris. Une première question est donc de savoir comment se comportent des réseaux profonds tels que SegNet pour la segmentation sémantique à partir de ces images. Par souci d'exhaustivité, signalons que des approches opérant directement sur le signal Lidar existent, mais ne font pas partie du cadre de cette étude. En particulier, Yan, Shaker et El-Ashmawy [64] s'intéressent à l'obtention de cartes d'occupation des sols par la classification automatique des échos Lidar, tandis que Yang et al. [65] utilisent un CNN pour la segmentation sémantique directement dans les nuages de points.

En ce qui nous concerne, nous considérons seulement les images en niveaux de gris correspondant aux MNE et aux MNH du jeu de données ISPRS Vaihingen. Nous réutilisons les hyperparamètres du Chapitre 3. L'objectif est de mesurer la quantité d'information présente au sein des modèles de terrain, qui peuvent alors être indifféremment dérivés du Lidar ou d'une paire d'images stéréo.

Nous entraînons donc un modèle SegNet à un seul canal sur le MNE et le MNH. Nous utilisons les tuiles 1, 3, 7, 11, 13, 17, 23, 26, 28, 32, 34 et 37 pour l'apprentissage et les tuiles 5, 15, 21 et 30 pour la validation. Les poids sont initialisés aléatoirement en utilisant la méthode de He et al. [26]. Le tableau Tableau 4.7 compile les scores F_1 obtenus pour les cinq classes d'intérêt du jeu de données ISPRS Vaihingen ainsi que l'exactitude globale du modèle. Ces résultats peuvent être comparés à ceux obtenus dans le Tableau 3.5 de la Section 3.3.4.

Tableau 4.7 – Résultats de validation sur le jeu de données ISPRS Vaihingen pour un modèle SegNet entraîné sur les MNE et MNH (scores F_1 et exactitude globale).

Entrée	Routes	Bâtiments	Vég. basse	Arbres	Véhicules	Exactitude
MNH	78,57	93,16	55,86	83,80	32,29	80,53
MNE	77,94	92,69	56,57	84,15	60,60	80,29
MNH + MNE	77,67	93,47	55,93	84,01	28,39	80,30

On constate que l'utilisation seule d'un des modèles numériques de terrain permet d'obtenir des scores F_1 élevés en inférence pour les routes, les bâtiments et les arbres. En effet, ces classes sont les plus simples à discriminer à partir de l'information de hauteur fournie par le Lidar. Les bâtiments présentent des surfaces surélevées régulières planes, les arbres sont des objets hauts à la surface chaotique et le sol est une large surface plane régulière à faible pente. Il est intéressant de constater que le modèle intègre un a priori spatial concernant la répartition de la végétation basse, qu'il place aléatoirement autour des arbres afin de créer des zones végétalisées. En outre, les véhicules sont prédits avec une précision relative à partir du MNH. Toutefois, le procédé de normalisation utilisé pour générer le MNH aplanit les zones appartenant au sol et les voitures y disparaissent généralement, provoquant une chute catastrophique des performances pour la classe des véhicules.

Néanmoins, malgré ce succès relatif, les modèles numériques de terrains obtiennent une exactitude globale nettement inférieure à celle obtenue à partir des images optiques IRRV.

4.3.2 Construction d'une image composite

Comme nous l'avons vu, les modèles numériques de terrain seuls ne suffisent pas pour couvrir la diversité des classes d'intérêt liées à la segmentation sémantique en zone urbaine. En effet, l'information qui y est contenue n'est pas suffisante pour pouvoir distinguer la végétation basse des surfaces imperméables, ni pour distinguer les véhicules dont la hauteur est trop faible pour être détectée de façon robuste à partir du MNH.

Couvrir toutes les classes nécessite ainsi une information de hauteur, mais également une information radiométrique. Le NDVI est un indice de végétation défini comme le rapport normalisé entre la réponse d'une surface dans le proche infrarouge et sa réponse dans le rouge :

$$NDVI = \frac{IR - \mathrm{R}}{IR + \mathrm{R}} \ . \tag{4.3}$$

Le NDVI prend des valeurs entre +1 et −1 indiquant respectivement une présence forte de végétation et une absence complète de végétation. Le NDVI est efficace car il modélise le pic de réponse de la végétation dans le proche-infrarouge et une absorption dans le spectre rouge dus à la présence de chlorophylle dans le feuillage. Ainsi, le NDVI permet de caractériser la présence et la densité de végétation présente sur la surface observée [42]. Le NDVI permet par ailleurs de caractériser des structures artificielles lorsqu'il est très faible [52].

On construit donc une image composite à trois canaux à partir du MNE, du MNH et du NDVI, telle qu'illustré dans la Figure 4.10.

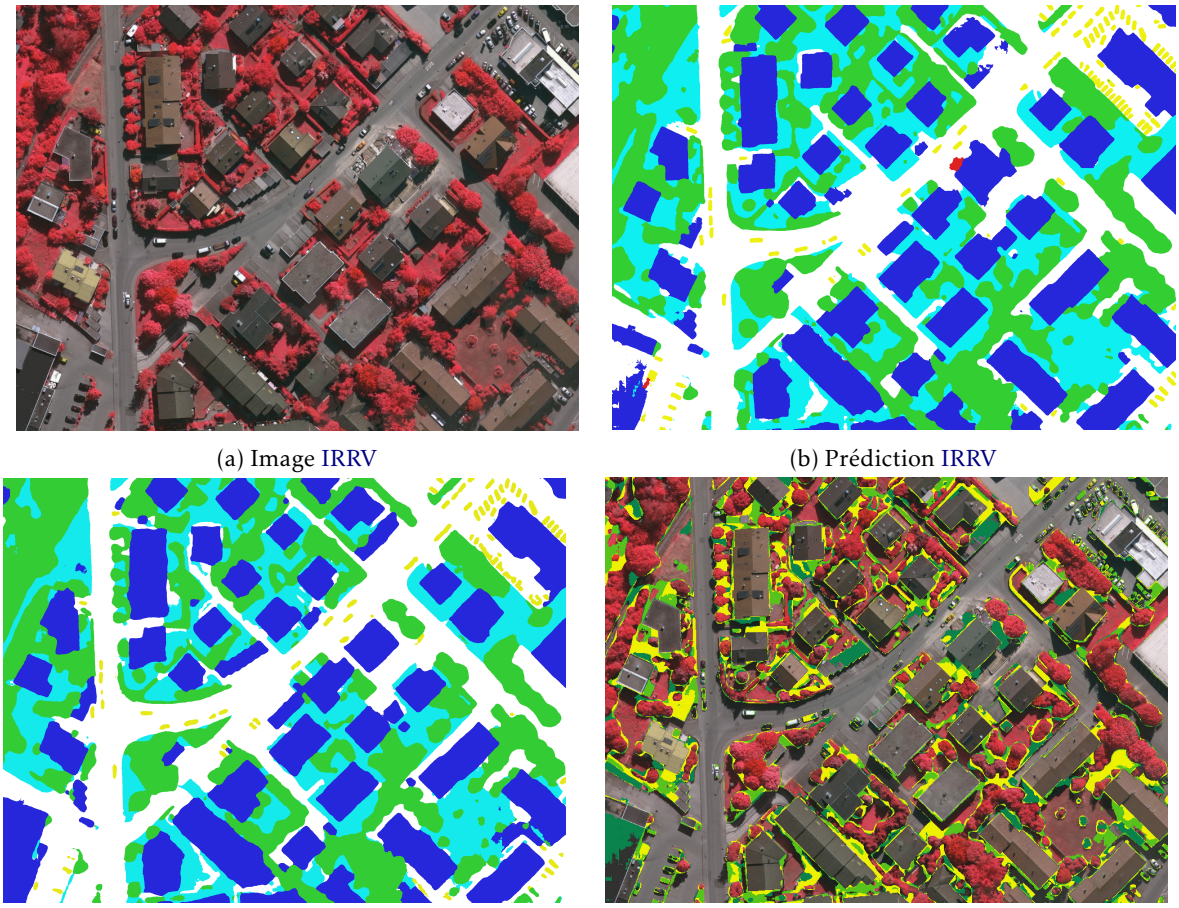
Tableau 4.8 – Résultats de validation sur le jeu de données ISPRS Vaihingen pour un modèle SegNet entraîné sur les images composites (avec et sans préentraînement sur ImageNet).

Entrée	Transfert	Routes	Bâtiments	Vég. basse	Arbres	Véhicules	Exactitude
Composite	✗	91,39	95,02	75,68	88,66	61,86	89,07
Composite	✓	91,34	95,48	76,47	89,39	73,47	89,61
IRRV	✓	91,43	95,37	79,97	90,53	90,41	90,47

Tableau 4.9 – Résultats de validation sur le jeu de données ISPRS Potsdam pour un modèle SegNet entraîné sur les images composites (avec et sans préentraînement sur ImageNet).

Entrée	Transfert	Routes	Bâtiments	Vég. basse	Arbres	Véhicules	Exactitude
Composite	✗	89,81	96,72	79,04	80,55	87,60	87,87
Composite	✓	90,81	97,23	80,89	81,17	92,47	89,20
RVB	✓	92,35	97,62	85,18	87,19	96,11	91,22

Les Tableaux 4.8 et 4.9 détaillent les résultats obtenus par un SegNet entraîné sur les images composite MNE/MNH/NDVI respectivement des jeux de données ISPRS Vaihingen et Potsdam. Les performances sont nettement supérieures à celles des modèles MNE et MNH pris séparément (cf. Tableau 4.7). En outre, la possibilité d'utiliser les poids préentraînés de VGG-16 sur ImageNet permet de gagner en exactitude sur l'ensemble des classes. Toutefois, la performance globale du modèle n'atteint pas celle de SegNet entraîné directement sur les images IRRV.



(a) Image IRRV (b) Prédiction IRRV

(c) Prédiction composite (d) Masque d'erreurs

Figure 4.11 – Différences entre les prédictions des modèles IRRV et composite. (b),(c) Légende : blanc : routes, bleu : bâtiments, cyan : végétation basse, vert : arbres, jaune : véhicules, rouge : autre. (d) En vert clair les erreurs du modèle entraîné en composite, en vert foncé les erreurs du modèle entraîné en IRRV et en jaune l'intersection des deux masques.

La Figure 4.11 illustre les prédictions obtenues via SegNet entraîné respectivement sur les données IRRV et sur les données composites. Le premier ne contient que 12% de pixels erronés tandis que le second est à environ 13% d'erreur. Cependant, il est remarquable que ces erreurs sont complémentaires, c'est-à-dire qu'elles ne se produisent pas sur les mêmes pixels. En effet, chaque modalité renseigne le modèle de différente façon. Si nous étions capables de fusionner parfaitement les deux cartes, de telle sorte que seuls les pixels qui sont classifiés de façon erronée dans les deux modèles soit en échec, alors le taux d'erreur tomberait à 7% sur cette image (masque jaune). Nous avons donc tout intérêt à étudier les possibilités d'apprentissage multimodal offertes par les réseaux profonds. C'est l'objet du chapitre suivant.

Les travaux présentés dans ce chapitre ont été le sujet de publications en conférence :

— Amina Ben Hamida et al. « Deep Learning for Semantic Segmentation of Remote

Sensing Images with Rich Spectral Content ». Dans : *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. Juil. 2017, p. 2569-2572. DOI : 10.1109/IGARSS.2017.8127520

— Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « Semantic Segmentation of Earth Observation Data Using Multimodal and Multi-Scale Deep Networks ». Dans : *Computer Vision – ACCV 2016*. Springer, Cham, 20 nov. 2016, p. 180-196. DOI : 10.1007/978-3-319-54181-5_12

Deux des publications conférences ont été étendues en article de journal :

— Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « Beyond RGB : Very High Resolution Urban Remote Sensing with Multimodal Deep Networks ». Dans : *ISPRS Journal of Photogrammetry and Remote Sensing* (23 nov. 2017). ISSN : 0924-2716. DOI : 10.1016/j.isprsjprs.2017.11.011

— Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « Deep Learning for Classification of Hyperspectral Data : A Comparative Review ». Dans : *IEEE Geoscience and Remote Sensing Magazine* in press (mar. 2019)

Références

[1] Olivier Arino et al. *Global Land Cover Map for 2009 (GlobCover 2009)*. 23 août 2012. DOI : https://doi.org/10.1594/PANGAEA.787668. URL : https://doi.pangaea.de/10.1594/PANGAEA.787668 (cf. p. 94).

[2] Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « Semantic Segmentation of Earth Observation Data Using Multimodal and Multi-Scale Deep Networks ». Dans : *Computer Vision – ACCV 2016*. Springer, Cham, 20 nov. 2016, p. 180-196. DOI : 10.1007/978-3-319-54181-5_12 (cf. p. 113).

[3] Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « Beyond RGB : Very High Resolution Urban Remote Sensing with Multimodal Deep Networks ». Dans : *ISPRS Journal of Photogrammetry and Remote Sensing* (23 nov. 2017). ISSN : 0924-2716. DOI : 10.1016/j.isprsjprs.2017.11.011 (cf. p. 113).

[4] Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « Deep Learning for Classification of Hyperspectral Data : A Comparative Review ». Dans : *IEEE Geoscience and Remote Sensing Magazine* in press (mar. 2019) (cf. p. 113).

[5] Amina Ben Hamida et al. « Deep Learning Approach for Remote Sensing Image Analysis ». Dans : *Big Data from Space (BiDS'16)*. Sous la dir. de SOILLE Pierre MARCHETTI Pier Giorgio. Santa Cruz de Tenerife, Spain : Publications Office of the European Union, mar. 2016, p. 133. DOI : 10.2788/854791. URL : https://hal.archives-ouvertes.fr/hal-01370161 (cf. p. 106-108).

[6] Amina Ben Hamida et al. « Deep Learning for Semantic Segmentation of Remote Sensing Images with Rich Spectral Content ». Dans : *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. Juil. 2017, p. 2569-2572. DOI : 10.1109/IGARSS.2017.8127520 (cf. p. 94, 112).

[7] Marco Bevilacqua et Yannick Berthoumieu. « Unsupervised Hyperspectral Band Selection via Multi-Feature Information-Maximization Clustering ». Dans : *2017 IEEE International Conference on Image Processing (ICIP)*. Pékin, China : IEEE, sept. 2017. DOI : 10.1109/ICIP.2017.8296339. URL : https://hal.archives-ouvertes.fr/hal-01717011 (cf. p. 103).

[8] Gustavo Camps-Valls et al. « Composite Kernels for Hyperspectral Image Classification ». Dans : *IEEE Geoscience and Remote Sensing Letters* 3.1 (2006), p. 93-97. URL : http://ieeexplore.ieee.org/abstract/document/1576697/ (cf. p. 104).

[9] Xavier Ceamanos et al. « Using 3D Information for Atmospheric Correction of Airborne Hyperspectral Images of Urban Areas ». Dans : *2017 Joint Urban Remote Sensing Event (JURSE)*. Mar. 2017, p. 1-4. DOI : 10.1109/JURSE.2017.7924563 (cf. p. 100).

[10] Pat S. Chavez. « Image-Based Atmospheric Corrections : Revisited and Improved ». Dans : *Photogrammetric engineering and remote sensing* 62.9 (1996), p. 1025-1036. URL : http://cat.inist.fr/?aModele=afficheN&cpsidt=3201162 (cf. p. 99).

[11] Yushi Chen, Xing Zhao et Xiuping Jia. « Spectral-Spatial Classification of Hyperspectral Data Based on Deep Belief Network ». Dans : *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 8.6 (juin 2015), p. 2381-2392. ISSN : 1939-1404, 2151-1535. DOI : 10.1109/JSTARS.2015.2388577. URL : http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7018910 (cf. p. 105).

[12] Yushi Chen et al. « Deep Learning-Based Classification of Hyperspectral Data ». Dans : *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 7.6 (juin 2014), p. 2094-2107. ISSN : 1939-1404. DOI : 10.1109/JSTARS.2014.2329330 (cf. p. 105).

[13] Yushi Chen et al. « Deep Feature Extraction and Classification of Hyperspectral Images Based on Convolutional Neural Networks ». Dans : *IEEE Transactions on Geoscience and Remote Sensing* 54.10 (oct. 2016), p. 6232-6251. ISSN : 0196-2892. DOI : 10.1109/TGRS.2016.2584107 (cf. p. 106).

[14] Ziyue Chen, Bingbo Gao et Bernard Devereux. « State-of-the-Art : DTM Generation Using Airborne LIDAR Data ». Dans : *Sensors* 17.1 (14 jan. 2017), p. 150. DOI : 10.3390/s17010150. URL : http://www.mdpi.com/1424-8220/17/1/150 (cf. p. 110).

[15] Manuel Cubero-Castan et al. « A Physics-Based Unmixing Method to Estimate Subpixel Temperatures on Mixed Pixels ». Dans : *IEEE Transactions on Geoscience and Remote Sensing* 53.4 (avr. 2015), p. 1894-1906. ISSN : 0196-2892. DOI : 10.1109/TGRS.2014.2350771 (cf. p. 98).

[16] Yanwei Cui, Laetitia Chapel et Sébastien Lefèvre. « Scalable Bag of Subpaths Kernel for Learning on Hierarchical Image Representations and Multi-Source Remote Sensing Data Classification ». Dans : *Remote Sensing* 9.3 (24 fév. 2017), p. 196. DOI : 10.3390/rs9030196. URL : http://www.mdpi.com/2072-4292/9/3/196 (cf. p. 104).

[17] Fabio Dell'Acqua et al. « Exploiting Spectral and Spatial Information in Hyperspectral Urban Data with High Resolution ». Dans : *IEEE Geoscience and Remote Sensing Letters* 1.4 (oct. 2004), p. 322-326. ISSN : 1545-598X. DOI : 10.1109/LGRS.2004.837009 (cf. p. 104).

[18] P. Y. Deschamps et T. Phulpin. « Atmospheric Correction of Infrared Measurements of Sea Surface Temperature Using Channels at 3.7, 11 and 12 µm ». Dans : *Boundary-Layer Meteorology* 18.2 (1er mar. 1980), p. 131-143. ISSN : 0006-8314, 1573-1472. DOI : 10.1007/BF00121320. URL : https://link.springer.com/article/10.1007/BF00121320 (cf. p. 99).

[19] Sophie Fabre, Xavier Briottet et Audrey Lesaignoux. « Estimation of Soil Moisture Content from the Spectral Reflectance of Bare Soils in the 0.4–2.5 µm Domain ». Dans : *Sensors* 15.2 (2 fév. 2015), p. 3262-3281. DOI : 10.3390/s150203262. URL : http://www.mdpi.com/1424-8220/15/2/3262 (cf. p. 98).

[20] Mathieu Fauvel, Jocelyn Chanussot et Jón Atli Benediktsson. « A Spatial-Spectral Kernel-Based Approach for the Classification of Remote-Sensing Images ». Dans : *Pattern Recogn.* 45.1 (jan. 2012), p. 381-392. ISSN : 0031-3203. DOI : 10.1016/j.patcog.2011.03.035. URL : http://dx.doi.org/10.1016/j.patcog.2011.03.035 (cf. p. 104).

[21] Mathieu Fauvel et al. « Advances in Spectral-Spatial Classification of Hyperspectral Images ». Dans : *Proceedings of the IEEE* 101.3 (mar. 2013), p. 652-675. ISSN : 0018-9219. DOI : 10.1109/JPROC.2012.2197589 (cf. p. 104).

[22] Python Software Foundation. *Python Language Reference*. https ://www.python.org/. URL : https://www.python.org/ (cf. p. 107).

[23] Qiongying Fu et al. « Semi-Supervised Classification of Hyperspectral Imagery Based on Stacked Autoencoders ». Dans : *Proceedings of the 8th Interational Conference on Digital Image Processing (ICDIP)*. T. 10033. 2016. DOI : 10.1117/12.2245011. URL : http://dx.doi.org/10.1117/12.2245011 (cf. p. 105).

[24] Bo-Cai Gao et al. « Atmospheric Correction Algorithms for Hyperspectral Remote Sensing Data of Land and Ocean ». Dans : *Remote Sensing of Environment* 113 (2009), S17-S24. URL : http://www.sciencedirect.com/science/article/pii/S0034425709000741 (cf. p. 99).

[25] Pradeep Goel et al. « Classification of Hyperspectral Data by Decision Trees and Artificial Neural Networks to Identify Weed Stress and Nitrogen Status of Corn ». Dans : *Computers and Electronics in Agriculture* 39.2 (mai 2003), p. 67-93. ISSN : 0168-1699. DOI : 10.1016/S0168-1699(03)00020-6. URL : http://www.sciencedirect.com/science/article/pii/S0168169903000206 (cf. p. 105).

[26] Kaiming He et al. « Delving Deep into Rectifiers : Surpassing Human-Level Performance on ImageNet Classification ». Dans : *Proceedings of the IEEE International Conference on Computer Vision*. IEEE International Conference on Computer Vision (ICCV). Déc. 2015, p. 1026-1034. DOI : 10.1109/ICCV.2015.123 (cf. p. 110).

[27] Mingyi He, Bo Li et Huahui Chen. « Multi-Scale 3D Deep Convolutional Neural Network for Hyperspectral Image Classification ». Dans : *2017 IEEE International Conference on Image Processing (ICIP)*. 2017 IEEE International Conference on Image Processing (ICIP). Sept. 2017, p. 3904-3908. DOI : 10.1109/ICIP.2017.8297014 (cf. p. 106).

[28] Wei Hu et al. « Deep Convolutional Neural Networks for Hyperspectral Image Classification ». Dans : *Journal of Sensors* 2015 (2015). DOI : 10.1155/2015/258619. URL : https://www.hindawi.com/journals/js/2015/258619/ (cf. p. 104, 105, 107, 108).

[29] Arnaud Le Bris et al. « Extraction of Optimal Spectral Bands Using Hierarchical Band Merging out of Hyperspectral Data ». Dans : *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XL-3/W3 (20 août 2015), p. 459-465. ISSN : 2194-9034. DOI : 10.5194/isprsarchives-XL-3-W3-459-2015. URL : http://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XL-3-W3/459/2015/ (cf. p. 103).

[30] Yann LeCun et al. « Gradient-Based Learning Applied to Document Recognition ». Dans : *Proceedings of the IEEE* 86.11 (nov. 1998), p. 2278-2324. ISSN : 0018-9219. DOI : 10.1109/5.726791 (cf. p. 106).

[31] Hyungtae Lee et Heesung Kwoon. « Contextual Deep CNN Based Hyperspectral Classification ». Dans : *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. IGARSS. Beijing, juil. 2016, p. 3322-3325. DOI : 10.1109/IGARSS.2016.7729859 (cf. p. 106).

[32] Tong Li, Junping Zhang et Ye Zhang. « Classification of Hyperspectral Image Based on Deep Belief Networks ». Dans : *Image Processing (ICIP), 2014 IEEE International Conference On*. IEEE, 2014, p. 5132-5136. URL : http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=7026039 (cf. p. 105).

[33] Ying Li, Haokui Zhang et Qiang Shen. « Spectral–Spatial Classification of Hyperspectral Imagery with 3D Convolutional Neural Network ». Dans : *Remote Sensing* 9.1 (13 jan. 2017), p. 67. doi : 10.3390/rs9010067. url : http://www.mdpi.com/2072-4292/9/1/67 (cf. p. 106-108).

[34] Zhouhan Lin et al. « Spectral-Spatial Classification of Hyperspectral Image Using Autoencoders ». Dans : *Information, Communications and Signal Processing (ICICS) 2013 9th International Conference On*. IEEE, 2013, p. 1-5. url : http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6782778 (cf. p. 105).

[35] Bing Liu et al. « A Semi-Supervised Convolutional Neural Network for Hyperspectral Image Classification ». Dans : *Remote Sensing Letters* 8.9 (2 sept. 2017), p. 839-848. issn : 2150-704X. doi : 10.1080/2150704X.2017.1331053. url : https://doi.org/10.1080/2150704X.2017.1331053 (cf. p. 106).

[36] Yanan Luo et al. « HSI-CNN : A Novel Convolution Neural Network for Hyperspectral Image ». Dans : (28 fév. 2018). arXiv : 1802.10478 [cs]. url : http://arxiv.org/abs/1802.10478 (cf. p. 106).

[37] Xiaorui Ma, Hongyu Wang et Jie Geng. « Spectral-Spatial Classification of Hyperspectral Image Based on Deep Auto-Encoder ». Dans : *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 9.9 (sept. 2016), p. 4073-4085. issn : 1939-1404. doi : 10.1109/JSTARS.2016.2517204 (cf. p. 105).

[38] Eliza Mace et al. « Overhead Detection : Beyond 8-Bits and RGB ». Dans : (7 août 2018). arXiv : 1808.02443 [cs]. url : http://arxiv.org/abs/1808.02443 (cf. p. 95).

[39] Konstantinos Makantasis et al. « Deep Supervised Learning for Hyperspectral Data Classification through Convolutional Neural Networks ». Dans : *Geoscience and Remote Sensing Symposium (IGARSS), 2015 IEEE International*. Geoscience and Remote Sensing Symposium (IGARSS), 2015 IEEE International. Juil. 2015, p. 4959-4962. doi : 10.1109/IGARSS.2015.7326945 (cf. p. 105, 106).

[40] Elamkulam Midhun et al. « Deep Model for Classification of Hyperspectral Image Using Restricted Boltzmann Machine ». Dans : *Proceedings of the 2014 International Conference on Interdisciplinary Advances in Applied Computing*. ICONIAAC '14. New York, NY, USA : ACM, 2014, 35 :1-35 :7. isbn : 978-1-4503-2908-8. doi : 10.1145/2660859.2660946. url : http://doi.acm.org/10.1145/2660859.2660946 (cf. p. 105).

[41] Lichao Mou, Pedram Ghamisi et Xiao Xiang Zhu. « Deep Recurrent Neural Networks for Hyperspectral Image Classification ». Dans : *IEEE Transactions on Geoscience and Remote Sensing* 55.7 (juil. 2017), p. 3639-3655. issn : 0196-2892. doi : 10.1109/TGRS.2016.2636241 (cf. p. 105, 107, 108).

[42] Ranga B. Myneni et al. « The Interpretation of Spectral Vegetation Indexes ». Dans : *IEEE Transactions on Geoscience and Remote Sensing* 33.2 (mar. 1995), p. 481-486. issn : 0196-2892. doi : 10.1109/36.377948 (cf. p. 111).

[43] Vinod Nair et Geoffrey E. Hinton. « Rectified Linear Units Improve Restricted Boltzmann Machines ». Dans : *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*. 2010, p. 807-814 (cf. p. 107).

[44] Lucas Parra et al. « Unmixing Hyperspectral Data ». Dans : *Proceedings of the 12th International Conference on Neural Information Processing Systems*. MIT Press, 1999, p. 942-948. url : http://dl.acm.org/citation.cfm?id=3009790 (cf. p. 103).

[45] Fabian Pedregosa et al. « Scikit-Learn : Machine Learning in Python ». Dans : *Journal of Machine Learning Research* 12 (Oct 2011), p. 2825-2830. issn : ISSN 1533-7928. url : http://www.jmlr.org/papers/v12/pedregosa11a.html (cf. p. 107).

[46] Antonio Plaza et al. « Spatial/Spectral Endmember Extraction by Multidimensional Morphological Operations ». Dans : *IEEE Transactions on Geoscience and Remote Sensing* 40.9 (sept. 2002), p. 2025-2041. ISSN : 0196-2892. DOI : 10.1109/TGRS.2002.802494 (cf. p. 104).

[47] *PyTorch : Tensors and Dynamic Neural Networks in Python with Strong GPU Acceleration*. http ://pytorch.org/. 2016-. URL : http://pytorch.org/ (cf. p. 95, 107).

[48] Hafizur Rahman et Gérard Dedieu. « SMAC : A Simplified Method for the Atmospheric Correction of Satellite Measurements in the Solar Spectrum ». Dans : *International Journal of Remote Sensing* 15.1 (1er jan. 1994), p. 123-143. ISSN : 0143-1161. DOI : 10.1080/01431169408954055. URL : http://dx.doi.org/10.1080/01431169408954055 (cf. p. 99).

[49] Frédéric Ratle, Gustau Camps-Valls et Jason Weston. « Semisupervised Neural Networks for Efficient Hyperspectral Image Classification ». Dans : *IEEE Transactions on Geoscience and Remote Sensing* 48.5 (mai 2010), p. 2271-2282. ISSN : 0196-2892. DOI : 10.1109/TGRS.2009.2037898 (cf. p. 105).

[50] Craig Rodarmel et Jie Shan. « Principal Component Analysis for Hyperspectral Image Classification ». Dans : *Surveying and Land Information Science* 62.2 (2002), p. 115. URL : http://search.proquest.com/openview/621b3a7187ca7f1dff4769113d396b20/1?pq-origsite=gscholar&cbl=27246 (cf. p. 103).

[51] Adriana Romero, Carlo Gatta et Gustau Camps-Valls. « Unsupervised Deep Feature Extraction for Remote Sensing Image Classification ». Dans : *IEEE Transactions on Geoscience and Remote Sensing* PP.99 (2015), p. 1-14. ISSN : 0196-2892. DOI : 10.1109/TGRS.2015.2478379 (cf. p. 106).

[52] Mitsuteru Sakamoto et al. « Automatic Detection of Damaged Area of Iran Earthquake by High-Resolution Satellite Imagery ». Dans : *IGARSS 2004. 2004 IEEE International Geoscience and Remote Sensing Symposium*. IGARSS 2004. 2004 IEEE International Geoscience and Remote Sensing Symposium. T. 2. Sept. 2004, 1418-1421 vol.2. DOI : 10.1109/IGARSS.2004.1368685 (cf. p. 111).

[53] Viktor Slavkovikj et al. « Hyperspectral Image Classification with Convolutional Neural Networks ». Dans : *Proceedings of the 23rd ACM International Conference on Multimedia*. ACM Press, 2015, p. 1159-1162. ISBN : 978-1-4503-3459-4. DOI : 10.1145/2733373.2806306. URL : http://dl.acm.org/citation.cfm?doid=2733373.2806306 (cf. p. 105).

[54] Nitish Srivastava et al. « Dropout : A Simple Way to Prevent Neural Networks from Overfitting ». Dans : *Journal of Machine Learning Research* 15 (2014), p. 1929-1958. URL : http://jmlr.org/papers/v15/srivastava14a.html (cf. p. 107).

[55] Chao Tao et al. « Unsupervised Spectral-Spatial Feature Learning With Stacked Sparse Autoencoder for Hyperspectral Imagery Classification ». Dans : *IEEE Geoscience and Remote Sensing Letters* 12.12 (déc. 2015), p. 2438-2442. ISSN : 1545-598X. DOI : 10.1109/LGRS.2015.2482520 (cf. p. 105).

[56] Yuliya Tarabalka, Jón Atli Benediktsson et Jocelyn Chanussot. « Spectral–Spatial Classification of Hyperspectral Imagery Based on Partitional Clustering Techniques ». Dans : *IEEE Transactions on Geoscience and Remote Sensing* 47.8 (2009), p. 2973-2987. URL : http://ieeexplore.ieee.org/abstract/document/4840429/ (cf. p. 104).

[57] Yuliya Tarabalka, Jocelyn Chanussot et Jon Atli Benediktsson. « Segmentation and Classification of Hyperspectral Images Using Watershed Transformation ». Dans : *Pattern Recognition* 43.7 (2010), p. 2367-2379. URL : http://www.sciencedirect.com/science/article/pii/S003132031000049X (cf. p. 104).

[58] Thierry Toutin. « Comparison of Stereo-Extracted DTM from Different High-Resolution Sensors : SPOT-5, EROS-a, IKONOS-II, and QuickBird ». Dans : *IEEE Transactions on Geoscience and Remote Sensing* 42.10 (oct. 2004), p. 2121-2129. ISSN : 0196-2892. DOI : 10.1109/TGRS.2004.834641 (cf. p. 110).

[59] Devis Tuia, Rémi Flamary et Nicolas Courty. « Multiclass Feature Learning for Hyperspectral Image Classification : Sparse and Hierarchical Solutions ». Dans : *ISPRS Journal of Photogrammetry and Remote Sensing* 105 (juil. 2015), p. 272-285. ISSN : 0924-2716. DOI : 10.1016/j.isprsjprs.2015.01.006. URL : http://www.sciencedirect.com/science/article/pii/S0924271615000234 (cf. p. 104).

[60] Devis Tuia, Claudio Persello et Lorenzo Bruzzone. « Domain Adaptation for the Classification of Remote Sensing Data : An Overview of Recent Advances ». Dans : *IEEE Geoscience and Remote Sensing Magazine* 4.2 (juin 2016), p. 41-57. ISSN : 2168-6831. DOI : 10.1109/MGRS.2016.2548504. URL : http://ieeexplore.ieee.org/document/7486184/ (cf. p. 108).

[61] Lizhe Wang et al. « Spectral–Spatial Multi-Feature-Based Deep Learning for Hyperspectral Remote Sensing Image Classification ». Dans : *Soft Computing* 21.1 (1er jan. 2017), p. 213-221. ISSN : 1432-7643, 1433-7479. DOI : 10.1007/s00500-016-2246-3. URL : https://link.springer.com/article/10.1007/s00500-016-2246-3 (cf. p. 105).

[62] Junfeng Wu et al. « Semi-Supervised Conditional Random Field for Hyperspectral Remote Sensing Image Classification ». Dans : *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS). Juil. 2016, p. 2614-2617. DOI : 10.1109/IGARSS.2016.7729675 (cf. p. 104).

[63] Chen Xing, Li Ma et Xiaoquan Yang. « Stacked Denoise Autoencoder Based Feature Extraction and Classification for Hyperspectral Images ». Dans : *Journal of Sensors* 2016 (30 nov. 2015), e3632943. ISSN : 1687-725X. DOI : 10.1155/2016/3632943. URL : https://www.hindawi.com/journals/js/2016/3632943/abs/ (cf. p. 105).

[64] Wai Yeung Yan, Ahmed Shaker et Nagwa El-Ashmawy. « Urban Land Cover Classification Using Airborne LiDAR Data : A Review ». Dans : *Remote Sensing of Environment* 158 (1er mar. 2015), p. 295-310. ISSN : 0034-4257. DOI : 10.1016/j.rse.2014.11.001. URL : http://www.sciencedirect.com/science/article/pii/S0034425714004374 (cf. p. 110).

[65] Zhishuang Yang et al. « A Convolutional Neural Network-Based 3D Semantic Labeling Method for ALS Point Clouds ». Dans : *Remote Sensing* 9.9 (10 sept. 2017), p. 936. DOI : 10.3390/rs9090936. URL : http://www.mdpi.com/2072-4292/9/9/936 (cf. p. 110).

[66] Jun Yue et al. « Spectral-Spatial Classification of Hyperspectral Images Using Deep Convolutional Neural Networks ». Dans : *Remote Sensing Letters* 6.6 (3 juin 2015), p. 468-477. ISSN : 2150-704X. DOI : 10.1080/2150704X.2015.1047045. URL : http://dx.doi.org/10.1080/2150704X.2015.1047045 (cf. p. 106).

[67] Wenzhi Zhao et Shihong Du. « Spectral-Spatial Feature Extraction for Hyperspectral Image Classification : A Dimension Reduction and Deep Learning Approach ». Dans : *IEEE Transactions on Geoscience and Remote Sensing* 54.8 (août 2016), p. 4544-4554. ISSN : 0196-2892. DOI : 10.1109/TGRS.2016.2543748 (cf. p. 106).

[68] Wenzhi Zhao et al. « On Combining Multiscale Deep Learning Features for the Classification of Hyperspectral Remote Sensing Imagery ». Dans : *International Journal of Remote Sensing* 36.13 (3 juil. 2015), p. 3368-3379. ISSN : 0143-1161. DOI : 10.1080/2150704X.2015.1062157. URL : http://dx.doi.org/10.1080/2150704X.2015.1062157 (cf. p. 106).

Chapitre

5 Segmentation sémantique multimodale

"I can see nothing," said I, handing it back to my friend.
"On the contrary, Watson, you can see everything. You fail, however, to reason from what you see. You are too timid in drawing your inferences."

— Arthur Conan Doyle (The Adventure of the Blue Carbuncle, 1892)

Sommaire

5.1 Apprentissage multimodal 120
5.1.1 Réseaux de neurones et apprentissage multimodal 120
5.1.2 Transposition à la télédétection 122
5.2 Fusion de modèles 123
5.2.1 Fusion par apprentissage 123
5.2.2 Mélanges de modèles 124
5.2.3 Résultats expérimentaux 126
5.3 Connaissances *a priori* 131
5.3.1 OpenStreetMap 131
5.3.2 Information *a priori* comme capteur virtuel 132
5.3.3 Architectures multimodales pour l'information géographique . . . 133

Résumé du chapitre :

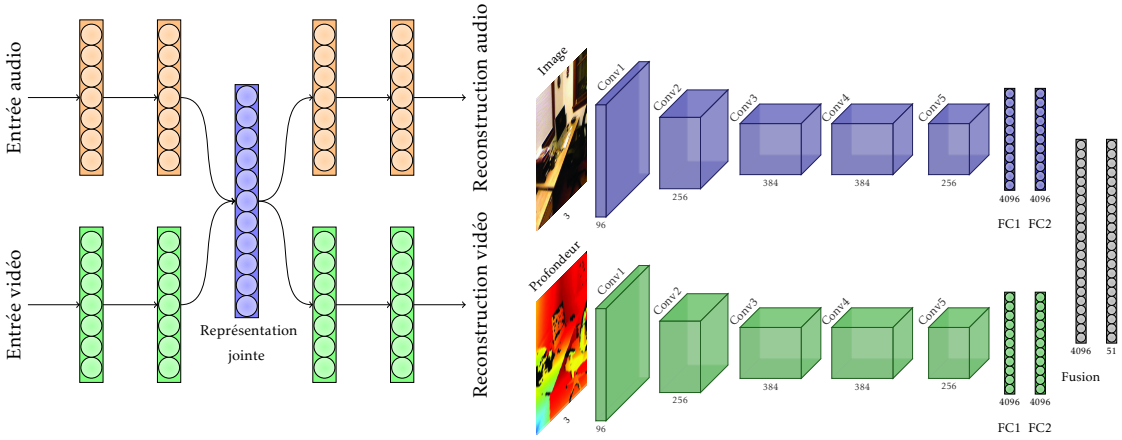
Les chapitres précédents nous ont permis d'établir de nouveaux états de l'art en segmentation sémantique d'images de télédétection sur l'ensemble des capteurs optiques couramment mis en œuvre pour l'observation de la Terre. Cependant, nous avons également observé l'insuffisance des modèles numériques de terrain dérivés du Lidar lorsqu'ils sont exploités seuls.

Dans ce chapitre, nous étudions les techniques de fusion de données permettant de combiner des informations issues de capteurs hétérogènes. Notamment, nous proposons plusieurs architectures multimodales pour l'apprentissage profond capables d'apprendre conjointement à partir des images optiques et des modèles numériques de terrain. Nous validons ces modèles avec succès sur plusieurs jeux de données et montrons ainsi qu'il est possible de tirer profit de la complémentarité des différents capteurs.

Nous étendons ensuite ces travaux à des sources de données non physiques. En particulier, nous appliquons ces architectures de fusion de donnée à des connaissances géographiques a priori provenant de bases de données en ligne, afin de consolider les cartes sémantiques prédites par le réseau. En considérant cette source d'information comme un capteur virtuel, les architectures multimodales précédemment développées nous permettent d'insérer des connaissances préalables dans les processus d'apprentissage et d'inférence de nos réseaux de neurones.

5.1 Apprentissage multimodal

5.1.1 Réseaux de neurones et apprentissage multimodal



(a) Auto-encodeur multimodal [30]

(b) CNN multimodal [9]

FIGURE 5.1 – Exemples d'architectures multimodales de réseaux profonds.

Les réseaux de neurones que nous avons présenté jusqu'ici n'opèrent que sur une unique source de données. Toutefois, la perception sensorielle humaine mobilise plusieurs modalités, notamment le son et l'image, présentant des interactions entre elles. Une question naturelle est donc de chercher à savoir dans quelle mesure une machine est en mesure d'exploiter ce type d'informations contenues sous plusieurs formes, partiellement redondantes.

Cette question relève de l'apprentissage multimodal et n'est pas spécifique au domaine de la télédétection. L'apprentissage de représentations à partir de signaux de natures hétérogènes est en effet un domaine de recherche à part entière. BALTRUŠAITIS, AHUJA et MORENCY [5] proposent la taxonomie suivante, dans laquelle les représentations peuvent être conjointes (une même représentation pour plusieurs modalités) ou coordonnées (chaque modalité à une représentation) :

- Représentation : génération de caractéristiques exploitant la complémentarité et la redondance de modalités multiples.
- Traduction : conversion d'une modalité à une autre.
- Alignement : identification de correspondances entre une représentation d'une modalité et une autre.
- Fusion : admission de plusieurs modalités pour la prise de décision.

Un modèle capable de synthétiser l'information visuelle et sonore d'une vidéo pour y assigner des étiquettes peut ainsi fonctionner de plusieurs façons. Une première possibilité est d'extraire des caractéristiques séparément pour le son et l'image puis d'utiliser un classifieur qui travaillera sur les caractéristiques concaténées (fusion). Il est également possible de contraindre les représentations audio et vidéo à respecter des critères de proximité sémantique (alignement) par des mesures de similarité. À l'inverse, une des modalités peut servir de référence à des algorithmes d'adaptation de domaine permettant de projeter une caractéristique audio vers une caractéristique image et inversement (traduction). Enfin, plutôt que de manipuler les représentations séparément, il est possible de construire un modèle travaillant directement sur la vidéo dans son hétérogénéité pour construire une représentation unique multimodale (représentation).

De nombreuses tâches sont liées à l'apprentissage multimodal, comme le sous-titrage automatique d'images [19], la classification de vidéos [20] ou la reconnaissance d'activités à partir de signaux issus de bracelets connectés [33]. De nombreux jeux de données multimodaux ont été proposés pour une grande variété d'applications : description automatique

d'images [15], diagnostic médical à partir de scanners variés [28], reconnaissance d'action à partir de données 3D et d'images [32], analyse d'émotions ressenties à partir de vidéos [36], éventuellement enrichie par des mesures cardiaques et neurales [35].

Une des premières méthodes développées pour l'apprentissage multimodal s'intéresse à la fusion dite tardive, intervenant lors de la prise de décision finale. Dans le cas le plus simple, il s'agit d'utiliser un modèle pour chaque modalité et d'appliquer des méthodes combinatoires d'apprentissage par ensemble, ou plus directement une combinaison linéaire, pour prendre la décision en tenant compte des entrées hétérogènes. C'est par exemple l'approche retenue par Yuhas, Goldstein et Sejnowski [45] et Meier, Hürst et Duchnowski [27] pour la reconnaissance automatique de syllabes à partir de vidéos. On retrouve régulièrement ce type d'approche dans la littérature récente pour le traitement de vidéos, par exemple dans [31] qui utilise une chaîne de Markov cachée pour la reconnaissance automatique de la parole.

Toutefois, l'apprentissage profond puise sa force dans l'expressivité des représentations que les modèles peuvent apprendre. Ainsi, Ngiam et al. [30] se sont par exemple intéressés à la construction d'un DBN auto-encodeur bi-modal audio-image. Deux encodeurs traitent chaque canal séparément et convergent vers une représentation partagée. Deux décodeurs doivent reconstruire chacun une des modalités à partir de la même représentation, comme illustré dans la Figure 5.1a. Un point particulièrement intéressant est que cette représentation partagée se montre capable de compenser la perte d'une modalité. Leur modèle peut ainsi être utilisé pour prédire un phonème à partir de la vidéo seulement ou de l'audio seulement. Srivastava et Salakhutdinov [42] proposent une architecture similaire conçue à partir de machines de Boltzmann profondes pouvant s'appliquer aux vidéos, mais aussi à des données très hétérogènes comme l'image (signal brut) et des étiquettes descriptives (langage symbolique). Leur modèle permet en outre de générer l'une des modalités à partir de l'autre lorsqu'elle est manquante. Plus récemment, Simonyan et Zisserman [39] ont introduit des réseaux à double flux pour la reconnaissance d'action dans des vidéos à partir des modalités son et image.

Récemment, l'émergence de capteurs RGB-D robustes et à faible coût comme le Kinect a encouragé la communauté vision à s'intéresser à la fusion entre images RVB et cartes de profondeur, c'est-à-dire aux données 2,5D. Ainsi, Couprie et al. [7] utilise un CNN pour l'extraction multiéchelle de caractéristiques sur des données RGB-D, considérées comme des images à 4 canaux. Toutefois, l'approche la plus courante consiste à concaténer des caractéristiques issues de modèles préentraînés pour générer des représentations multimodales artificielles [37, 21]. Afin d'en augmenter l'expressivité, il semble ainsi intéressant de chercher à automatiquement apprendre des représentations conjointes exploitant la complémentarité des sources de données. Eitel et al. [9], Guo, Wang et Chen [12] et Song, Jiang et Herranz [41] s'inspirent de fait du modèle de Ngiam et al. [30] en utilisant deux CNN en parallèle extrayant des caractéristiques qui sont fusionnées dans une représentation conjointe par les dernières couches, leur permettant ainsi de réaliser directement des classifications d'images RGB-D. Cette architecture est illustrée dans la Figure 5.1b. Concrètement, deux réseaux AlexNet en parallèle sont utilisés pour extraire des caractéristiques sur l'image RVB et sur une carte de profondeur encodée sur 3 canaux. Les caractéristiques extraites par les couches convolutives d'AlexNet convergent de la même façon que chez Ngiam et al. [30] et sont utilisées comme entrée du classifieur entièrement connecté commun aux deux réseaux. Ces approches permettent d'améliorer les performances de classifieurs convolutifs par rapport au travail sur l'image RVB seule, l'information de profondeur introduisant de l'*a priori* géométrique et diminuant l'influence des occlusions.

L'architecture FuseNet introduite par Hazirbas et al. [13] est une extension naturelle de cette approche à la segmentation sémantique. Appliqué sur des images RGB-D, FuseNet dérive du modèle SegNet [4] présenté dans le Chapitre 3. Deux encodeurs réalisent une extraction de caractéristiques dense sur l'image RVB et la carte de profondeur encodée sur 3

canaux. Un décodeur unique réalise le surréchantillonnage et la classification en simultané. Cette méthode permet à Hazirbas et al. [13] d'établir un nouvel état de l'art sur le jeu de données SUN RGB-D, dédié à la segmentation sémantique d'images RGB-D en intérieur. Guerry, Le Saux et Filliat [11] obtiennent également d'excellents résultats en détection de personnes dans des images RGB-D en utilisant ces mécanismes d'apprentissage bi-modaux au sein desquels les encodeurs parallèles s'échangent de l'information issue de capteurs complémentaires. Finalement, Lee, Park et Hong [22] proposent une amélioration de FuseNet en introduisant l'apprentissage résiduel en son sein, établissent à leur tour un nouvel état de l'art sur le jeu de données SUN RGB-D.

De cette revue de l'état de l'art, observons d'ores et déjà que les méthodes d'interprétation d'images RGB-D traitent toutes séparément les aspects de couleur et de profondeur. En effet, comme pour le multispectral, il est intéressant de pouvoir bénéficier des modèles préentraînés forts sur les images RVB. Concaténer la carte de profondeur à l'image couleur pour former un tenseur à 4 canaux résulte ainsi en des performances plus faibles qu'en traitant les données séparément.

5.1.2 Transposition à la télédétection

En télédétection, l'utilisation des modèles numériques de terrain pour améliorer les performances des classifieurs image a également été étudiée à plusieurs reprises. Le problème est en effet proche de l'interprétation d'images 2,5D RGB-D, le MNT jouant un rôle comparable à celui des cartes de profondeur.

Toutefois, la plupart des travaux antérieurs à cette thèse ont utilisé des stratégies *ad hoc* pour réaliser cette fusion. Lagrange et al. [21] concatènent simplement les caractéristiques extraites par des réseaux préentraînés pour optimiser un nouveau classifieur SVM et Paisitkriangkrai et al. [34] font de même avec des forêts aléatoires en y intégrant en sus des caractéristiques expertes. Plus récemment, Liu et al. [24] utilisent ces mêmes caractéristiques mais les injectent dans un modèle graphique de type CRF pour combiner segmentation sémantique issue d'un FCN, modèle de terrain et NDVI.

Dans cette thèse, nous étudions des approches profondes de bout en bout n'utilisant ni modèle graphique, ni caractéristiques *ad hoc*. Dans un premier temps, nous nous intéresserons à la fusion optique/modèle numérique de terrain, puis nous chercherons comment utiliser ces méthodes pour l'intégration de connaissances *a priori*.

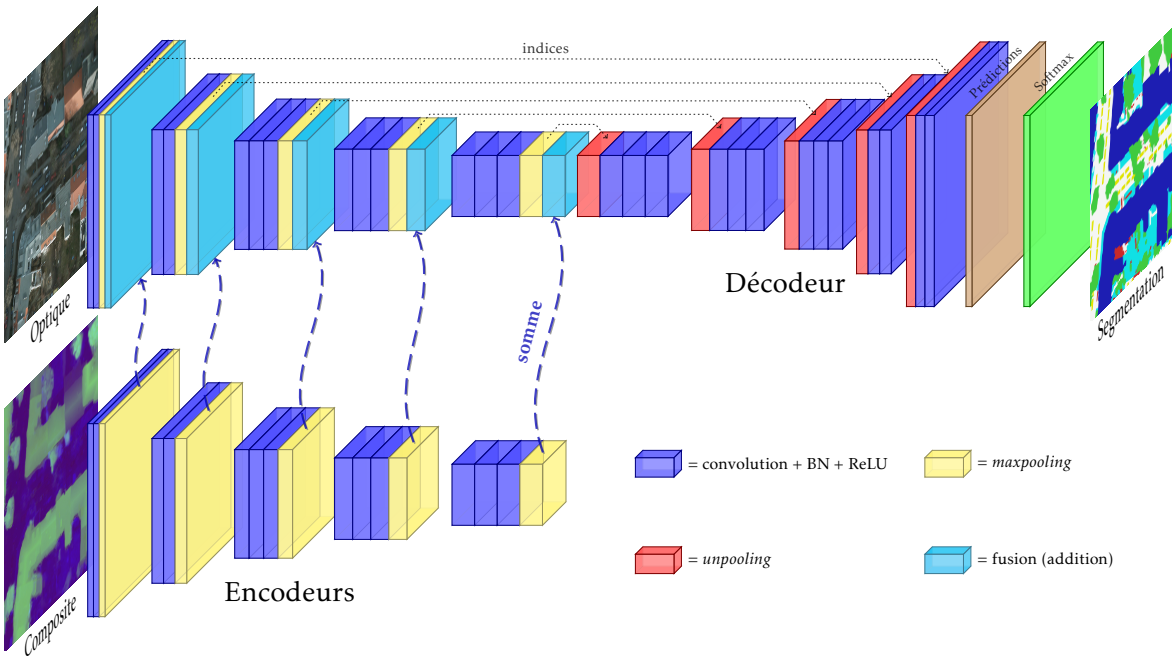


FIGURE 5.2 – Architecture FuseNet [13].

5.2 Fusion de modèles

5.2.1 Fusion par apprentissage

L'architecture FuseNet [13] est une variante multimodale de SegNet. Comme illustré dans la Figure 5.2, FuseNet encode conjointement l'image RVB et la carte de profondeur en utilisant deux encodeurs dont les contributions respectives sont additionnées après chaque bloc convolutif. Un décodeur unique réalise alors la reconstruction de la résolution et la classification. Cette approche peut être adaptée à n'importe quel autre CNN de base, comme les ResNet.

Plus formellement, en notant $\hat{\mathrm{P}}$ la fonction de prédiction modélisée par FuseNet appliquée à l'image I et la profondeur Δ, $\mathcal{D}$ le décodeur et $\mathrm{E}_i^{\mathrm{I}}, \mathrm{E}_i^{\Delta}$ les sorties du i^{e} bloc des encodeurs pour l'image et la profondeur et $\mathcal{B}_i$ l'opérateur modélisant le i^{e} bloc, alors :

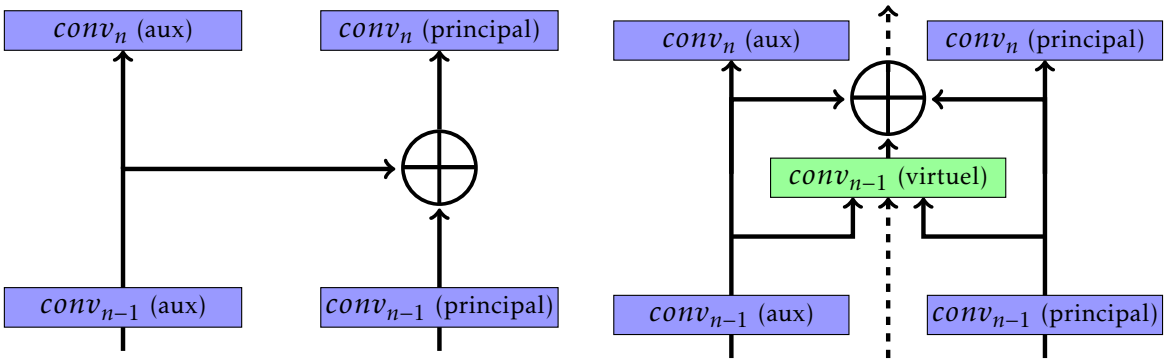
$$\hat{\mathrm{P}}(\mathrm{I},\Delta) = \mathcal{D}\left(\mathrm{E}_5^{\mathrm{I}}(\mathrm{I},\Delta)\right) \tag{5.1}$$

où

$$\begin{cases} \mathrm{E}_{i+1}^{\mathrm{I}}(\mathrm{I},\Delta) = \mathcal{B}_i^{I}\left(\mathrm{E}_i^{\mathrm{I}} + \mathrm{E}_i^{\Delta}\right) \\ \mathrm{E}_{i+1}^{\Delta}(\Delta) = \mathcal{B}_i^{\Delta}(\mathrm{E}_i^{\Delta}) \end{cases} \tag{5.2}$$

Dans notre cas, nous pouvons altérer FuseNet de la même façon que nous avons altéré SegNet dans les chapitres précédents, afin de traiter des images de télédétection. En effet, une carte d'élévation comme le MNE peut être considérée comme une carte de profondeur associée à une image RVB. De fait, nous proposons ainsi d'adapter FuseNet au traitement d'images de télédétection multimodales. En pratique, nous utiliserons comme sources d'entrées les images optiques, RVB ou IRRV, et les images composites générées dans la Section 4.3.2.

Néanmoins, l'architecture FuseNet considère les données de profondeur comme secondaires. En effet, les deux branches de l'encodeur ne sont pas complètement symétriques : la branche de profondeur ne considère que l'information de profondeur tandis que la branche optique opère en réalité sur une représentation conjointe RGB-D. De plus, le processus de sur-échantillonnage du décodeur ne considère que les indices de la branche principale,



(a) FuseNet original : les activations auxiliaires sont sommées dans la branche principale.

(b) V-FuseNet : les activations des deux branches sont fusionnées par un bloc convolutif résiduel.

Figure 5.3 – Stratégies de fusion pour l'architecture FuseNet.

c'est-à-dire la branche optique. Il est donc nécessaire de choisir une source principale de données et une source auxiliaire (cf. Figure 5.3a). Il y a donc un déséquilibre fondamental entre les traitements appliqués aux deux sources. Nous proposons une architecture alternative utilisant une troisième source de données virtuelle qui permet de faire disparaître cette asymétrie.

Plutôt que de calculer la somme des cartes d'activations, nous utilisons un processus de fusion permettant d'encoder des caractéristiques multimodales. Nous introduisons un troisième encodeur qui ne correspond à aucune modalité réelle, mais traite la représentation conjointe des données. Après le n^e bloc, l'encodeur virtuel concatène les activations des deux encodeurs réels et les transmet à un bloc convolutif résiduel, qui en génère une représentation multimodale conjointe. Ce sont ces caractéristiques qui sont ensuite décodées et sur-échantillonnées par la deuxième partie du modèle. Ce procédé est illustré par la Figure 5.3b. Cette stratégie permet de symétriser FuseNet et de ne plus avoir à choisir de source principale. Dans la taxonomie de Baltrušaitis, Ahuja et Morency [5], cela correspond au passage d'une méthode d'alignement à une approche par représentation. Cette architecture est nommée *V-FuseNet* dans le reste du chapitre.

Un autre inconvénient de FuseNet est que cette approche nécessite d'avoir des modèles de base topologiquement compatibles afin de pouvoir sommer les activations et fusionner les encodeurs. Cependant, cela ne se vérifie pas systématiquement. Par exemple, certaines données peuvent posséder des natures différentes, comme une image 2D et un nuage de point 3D. En outre, il n'est pas nécessairement utile de consacrer autant de paramètres aux deux sources de données, notamment si l'une est moins informative que l'autre. Nous proposons donc une méthode de fusion de données alternative pour extraire de l'information à partir de sources hétérogènes. En l'occurrence, nous suggérons d'étudier la possibilité d'effectuer une fusion tardive au moment de la prise de décision, plutôt que d'apprendre des représentations conjointes.

5.2.2 Mélanges de modèles

Une approche alternative consiste à séparer les traitements appliqués aux différentes modalités et à combiner les prédictions de l'ensemble des modèles. Ainsi, il est envisageable d'entraîner un réseau par modalité et de réaliser la moyenne des prédictions. Toutefois, cela ne permet pas de prendre en compte les particularités de chaque capteur. Nous introduisons donc un module de correction résiduelle qui prend en entrée les dernières cartes d'activation des réseaux et réalise une fusion des cartes de probabilités [1]. Le module de correction résiduelle apprend la correction ϵ à appliquer à la prédiction moyenne pour améliorer les performances globales du modèle combiné. Ce processus est illustré dans la Figure 5.4 pour SegNet.

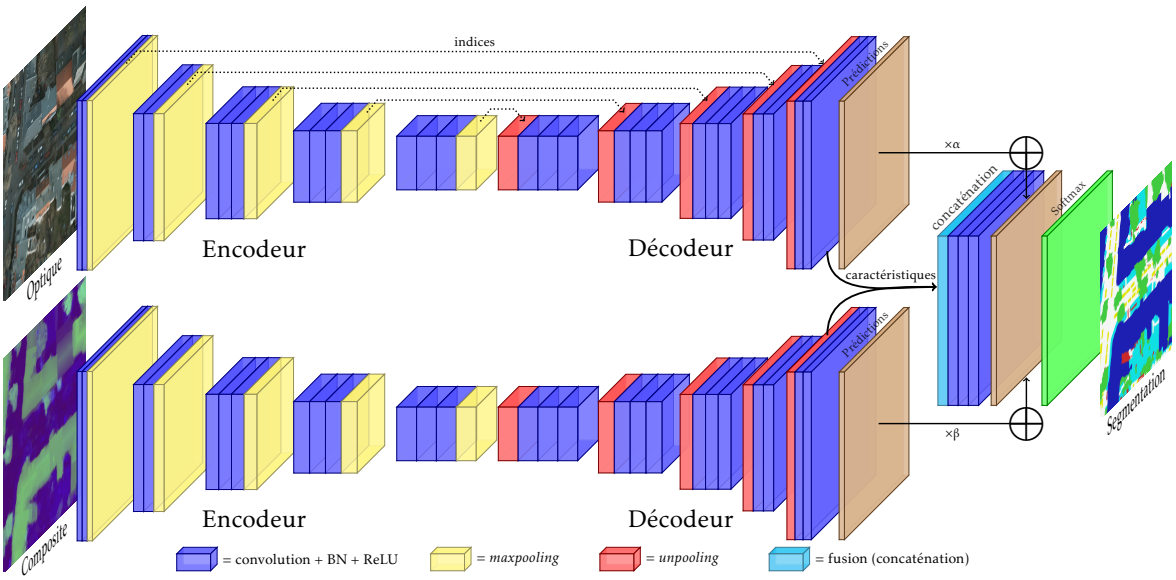


FIGURE 5.4 – Correction résiduelle appliquée à SegNet.

Ce module réalise une fusion au niveau décisionnel en utilisant le principe de l'apprentissage résiduel [14]. Il comporte trois couches convolutives de noyau 3×3 et un *padding* de 1 px. Les cartes d'activation intermédiaires des deux décodeurs de SegNet sont concaténées et utilisées comme entrées pour le module de correction (cf. Figure 5.4). La sortie du module est sommée de façon résiduelle avec la moyenne de prédictions issues des SegNet, comme illustré dans la Figure 5.5. Dans ce cas, l'apprentissage par résidu est particulièrement adapté car la prédiction moyenne peut déjà être considérée comme proche du résultat visé. Le module additionnel vient donc fusionner les décisions en appliquant une moyenne pondérée adaptative dépendante des cartes d'activations afin d'ajouter un terme correctif à la prédiction moyenne. Ce module, entraîné par rétropropagation, est simplement optimisé *a posteriori* par *fine-tuning*, les SegNet n'étant pas ré-entraînés. Ce procédé est donc rapide, car seuls les gradients sur les couches du module correctif sont calculés. Il est possible d'entraîner l'ensemble de bout en bout, mais cela nécessite alors de stocker l'ensemble des gradients en mémoire, ce qui n'est pas nécessairement possible sur tous les GPU. L'approche multimodale SegNet avec correction résiduelle est notée *SegNet-CR* par la suite.

Notons $\mathrm{P}_{réelle}$ le tenseur représentant la vérité terrain et $\hat{\mathrm{P}}_i$ les prédictions réalisées par la i^{e} sortie. On définit alors le terme d'erreur ϵ_i tel que :

$$\hat{\mathrm{P}}_i = \mathrm{P}_{réelle} + \epsilon_i \quad \text{avec} \quad |\epsilon_i| \ll |\hat{\mathrm{P}}_i| \ . \tag{5.3}$$

Si la prédiction P_i est proche de la vérité terrain, alors ϵ_i reste faible. L'objectif du module de correction résiduelle est d'apprendre à estimer l'erreur afin de pouvoir la corriger lors de l'inférence.

En notant n le nombre de prédictions à fusionner par correction résiduelle, alors la sortie du module notée $\hat{\mathrm{P}}^*$ correspond à la somme de la prédiction moyenne des $\hat{\mathrm{P}}_i$ et d'un terme correcteur c :

$$\hat{\mathrm{P}}^* = \hat{\mathrm{P}}_{moyenne} + c = \frac{1}{n}\sum_{i=1}^{n} \mathrm{P}_i + c = \mathrm{P}_{réelle} + \frac{1}{n}\sum_{i=1}^{n} \epsilon_i + c \ . \tag{5.4}$$

Le module de correction résiduelle étant optimisé pour minimiser la fonction de coût, cette contrainte se traduit par :

$$\left\| \hat{\mathrm{P}}^* - \mathrm{P}_{réelle} \right\| \to 0 \tag{5.5}$$

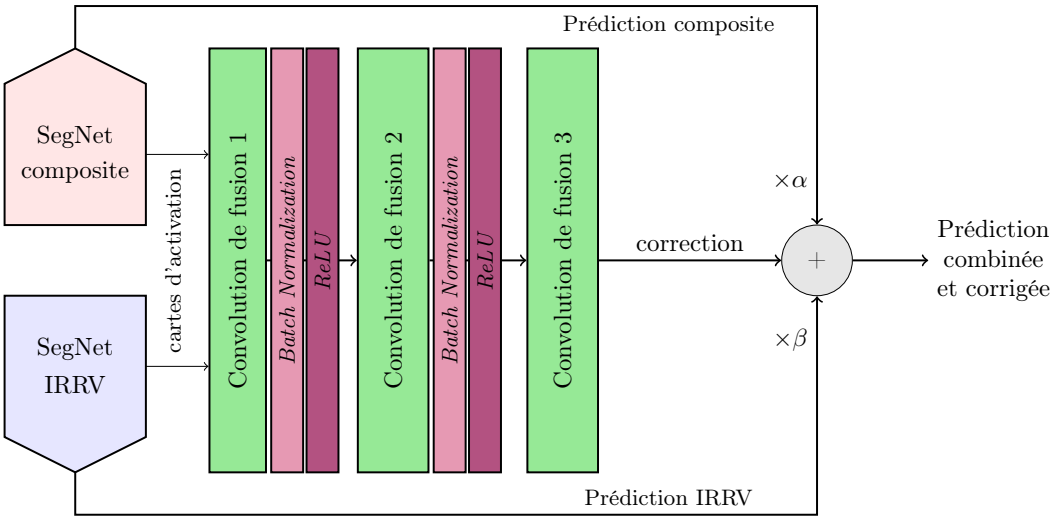


FIGURE 5.5 – Module de correction résiduelle.

ce qui impose en retour la contrainte suivante sur c et ϵ_i :

$$\left\| \frac{1}{n} \sum_{i=1}^{n} \epsilon_i - c \right\| \to 0 \ . \tag{5.6}$$

Autrement dit, le module de correction résiduelle est optimisé afin de compenser l'erreur moyenne commise par les différents modèles de l'ensemble. Lors de la phase d'entraînement, la vérité terrain $P_{réelle}$ est connue. Les poids du module sont alors altérés par rétro-propagation de sorte que le terme correctif c se rapproche de $\frac{1}{n}\sum_{i=1}^{n}\epsilon_i$. L'erreur moyenne étant supposée faible, la correction d'erreur correspond ainsi au paradigme d'apprentissage par résidu [14]. En effet, c est un terme additif de faible amplitude sommé au signal initial (ou *bypass*). Cette approche est schématisée dans la Figure 5.5.

5.2.3 Résultats expérimentaux

TABLEAU 5.1 – Résultats de segmentation sémantique multimodale sur le jeu de validation ISPRS Vaihingen.

Modèle	Exactitude	Score F_1
SegNet (IRRV)	90,2± 1,4	89,3± 1,2
SegNet (composite)	88,3± 0,9	81,6± 0,8
SegNet-CR	90,6± 1,4	89,2± 1,2
FuseNet	90,8± 1,4	90,1± 1,2
V-FuseNet	**91,1**± 1,5	**90,3**± 1,2
ResNet-34 (IRRV)	90,3± 1,0	89,1± 0,7
ResNet-34 (composite)	88,8± 1,1	83,4± 1,3
ResNet-34-CR	90,8± 1,0	89,1± 1,1
FusResNet	90,6± 1,1	89,3± 0,7

Comme attendu, les deux méthodes de fusion permettent d'améliorer les performances du classifieur sur les deux jeux de données, comme illustré par les Figures 5.6a et 5.7 et les Tableaux 5.1 à 5.3. Comme dans le Chapitre 3, utiliser ResNet-34 comme modèle de base plutôt que SegNet n'améliore que légèrement les performances et ne justifie pas le surcoût en temps de calcul que cela implique. En particulier, la Figure 5.6 montre plusieurs

Tableau 5.2 – Résultats de segmentation sémantique multimodale sur le jeu de test ISPRS Vaihingen (approches multimodales). Les meilleurs résultats sont en **gras** et les seconds sont en *italique*.

Modèle	Routes	Bâtiments	Vég. basse	Arbres	Véhicules	Exactitude
FCN+CRF + contours + MNH corrigé [26]	92,4	95,2	83,9	89,9	81,2	90,3
SegNet (IRRV)	91,5	94,3	82,7	89,3	85,7	89,4
SegNet-CR	91,0	94,5	84,4	89,9	77,8	89,8
FuseNet	91,3	94,3	84,8	89,9	85,9	90,1
V-FuseNet	91,0	94,4	84,5	89,9	86,3	90,0

Tableau 5.3 – Résultats de segmentation sémantique multimodale sur le jeu de test ISPRS Potsdam (approches multimodales). Les meilleurs résultats sont en **gras** et les seconds sont en *italique*.

Modèle	Routes	Bâtiments	Vég. basse	Arbres	Véhicules	Exactitude
FCN + CRF + caractéristiques expertes [24]	91,2	94,6	85,1	85,1	92,8	88,4
FCN [38]	92,5	96,4	86,7	88,0	94,7	90,3
SegNet (IRRV)	92,4	95,8	86,7	87,4	95,1	90,0
SegNet-CR	**93,3**	**97,3**	*87,6*	**88,3**	**95,8**	**91,0**
FuseNet	93,0	97,0	87,3	87,7	*95,2*	*90,6*
V-FuseNet	*93,2*	*97,2*	**87,9**	*88,2*	95,0	**91,0**

exemples d'objets erronément classifiés à partir de l'image optique seule qui sont corrigés en intégrant l'image composite dans le modèle. Dans les Figures 5.6a et 5.6b, le réseau SegNet ne parvient pas à discriminer entre les classes de route et de bâtiment. En effet, l'apparence du parking sur le toit est similaire à celle des zones de stationnement habituellement, confusion renforcée par la présence des voitures et des lignes blanches. FuseNet utilise le MNH pour trancher en faveur de la classe bâtiment et ignore de fait les véhicules. L'approche par correction résiduelle parvient en plus à conserver une partie de l'information spatiale liée aux voitures. Dans la Figure 5.6c, SegNet confond également route et bâtiment et végétation haute et basse tandis que les deux architectures de fusion parviennent à correctement prédire les différentes objets grâce au MNH. La fusion dans l'encodeur par FuseNet permet d'exploiter conjointement les modalités multiples en utilisant moins de paramètres que la correction résiduelle d'un ensemble de modèles et converge vers une meilleure performance globale. À l'opposé, la fusion tardive par correction résiduelle améliore les performances de manière moins équilibrée, les gains étant principalement concentrés sur les classes "routes" et "bâtiments".

Un des avantages pratique de la correction résiduelle est de pouvoir fusionner des prédictions en fonction de l'intensité des activations. La Figure 5.6b donne ainsi un exemple de fusion réussie, dans laquelle la confusion du modèle IRRV autour des voitures est compensée par la confiance élevée de la prédiction du modèle composite. En outre, par rapport à la simple moyenne adaptative des deux modèles, l'introduction du module de correction résiduelle permet de faire progresser l'exactitude globale de 0,4%, justifiant de fait l'intérêt d'une telle approche par raffinement.

L'architecture FuseNet apprend une représentation conjointe des deux sources de données, mais fait face aux même problématiques que le modèle SegNet standard : les cas rares comme les voitures sur un toit sont incorrectement prédits. Les caractéristiques conjointes apprises par les deux encodeurs sont plus performantes, mais FuseNet reste sensible au sur-apprentissage et aux biais intrinsèques du jeu de données, là où la correction résiduelle

parvient à corriger des erreurs même sur des cas rares. La fusion tardive est ainsi pertinente lorsqu'il s'agit de combiner des prédictions fortement complémentaires et agit comme une moyenne pondérée adaptative des prédictions. Pour les cas du parking aérien, la prédiction du SegNet composite prédit un bâtiment avec une confiance haute car le MNH est fiable, tandis que le SegNet RVB produit des prédictions à fort niveau de confiance sur les voitures. À l'inverse, FuseNet apprend des représentations conjointes qui n'échappent pas au surapprentissage. Ainsi, les voitures sur le parking aérien disparaissent à l'inférence car c'est un cas unique au sein du jeu de données : les voitures sont normalement sur la route. En conclusion, ces deux stratégies de fusion sont applicables à différents cas d'utilisation. La fusion tardive par correction résiduelle est utile pour combiner des classifieurs forts très complémentaires, tandis que la stratégie FuseNet est plus adaptée pour exploiter de l'information annexe ancillaire dans le processus d'apprentissage. Sur le jeu de test final de Vaihingen (cf. Tableau 5.2), la stratégie V-FuseNet a des performances marginalement inférieures au modèle original, bien que les F_1 scores soient supérieurs sur certaines classes, y compris la classe de rejet (+1,7%) qui n'est pas prise en compte dans les métriques finales. Sur le jeu de test ISPRS Potsdam, V-FuseNet est toutefois significativement plus performante aussi bien sur les classes individuelles que dans l'ensemble.

Robustesse aux données manquantes

Si ces architectures multimodales permettent de bénéficier de données géoréférencées recalées hétérogènes, elles introduisent néanmoins une limitation additionnelle. En effet, bien que les données Lidar du jeu de données ISPRS soient très denses, la normalisation du modèle de terrain est imparfaite et quelques artefacts ont été générés. Comme signalé par Marmanis et al. [26], plusieurs bâtiments sont absents du MNH, une hauteur de 0 m ayant été attribuée aux pixels correspondants. Cela cause des problèmes significatifs de classification pour les deux méthodes de fusion, comme illustré par la Figure 5.8. La solution proposée par [26] consiste à manuellement corriger le MNH, mais cette méthode ne passe pas à l'échelle. Des stratégies robustes à la perte d'une modalité ou aux données bruitées pourraient permettre de résoudre ce problème, en utilisant par exemple les *hallucination networks* de Hoffman, Gupta et Darrell [16] pour inférer les données manquantes [18]. Les travaux récents sur les modèles génératifs pourraient également permettre de diminuer le sur-apprentissage et améliorer la robustesse des modèles en entraînant ceux-ci sur des données synthétiques bruitées [44].

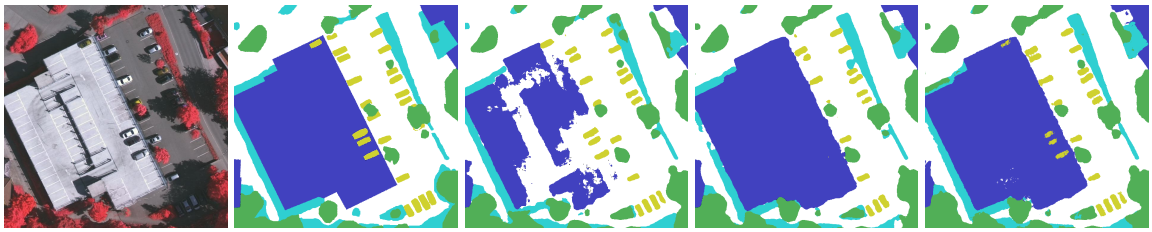
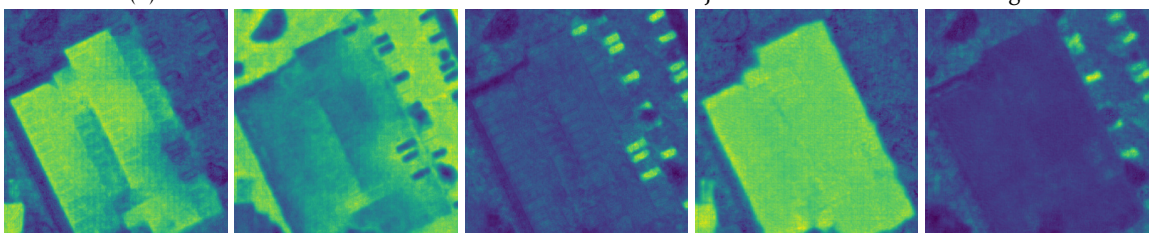


Image IRRV | Vérité terrain | SegNet | FuseNet | SegNet-CR

(a) Prédictions de différents modèles sur un extrait du jeu de données ISPRS Vaihingen.



SegNet IRRV (confiance, bâtiments) | SegNet IRRV (confiance, routes) | SegNet IRRV (confiance, voitures) | SegNet composite (confiance, bâtiments) | SegNet composite (confiance, voitures)

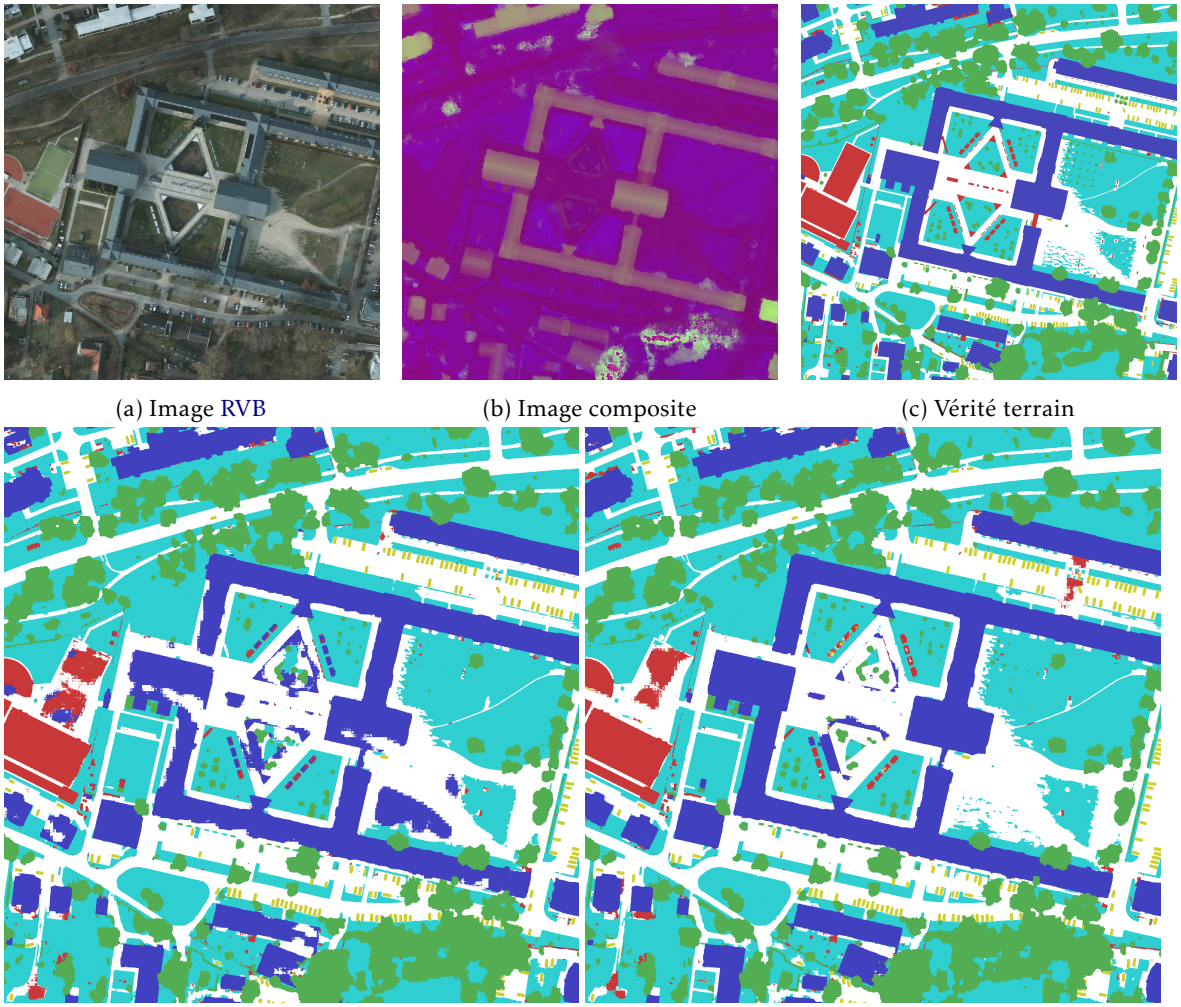
(b) Cartes de confiance de SegNet pour diverses classes selon le type d'entrée.



Image IRRV | Vérité terrain | SegNet | FuseNet | SegNet-CR

(c) Prédictions de différents modèles sur un extrait du jeu de données ISPRS Vaihingen.

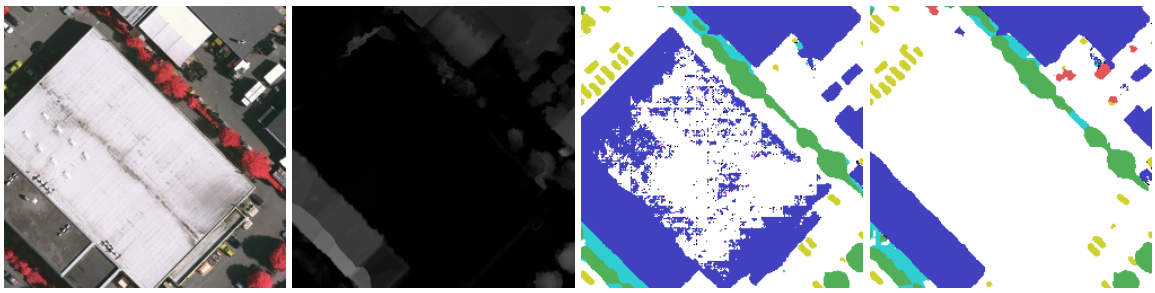
FIGURE 5.6 – Exemples de prédictions multimodales réussies sur Vaihingen.
Légende : blanc : routes, bleu : bâtiments, cyan : végétation basse, vert : arbres, jaune : véhicules, rouge : autre.



(a) Image RVB (b) Image composite (c) Vérité terrain

(d) SegNet (e) V-FuseNet

FIGURE 5.7 – Effet des stratégies de fusion sur un extrait du jeu de données ISPRS Potsdam. La confusion entre les classes de route et de bâtiments est nettement réduite grâce à la contribution des modèles de terrain.
Légende : blanc : routes, bleu : bâtiments, cyan : végétation basse, vert : arbres, jaune : véhicules, rouge : autre.



IRRV MNH SegNet-CR V-FuseNet

FIGURE 5.8 – Les erreurs dans le MNH du jeu de données ISPRS Vaihingen sont mal gérées par les deux méthodes de fusion. Ici, un bâtiment entier disparaît.

5.3 Connaissances *a priori*

La section précédente nous a permis de construire des architectures profondes multimodales pour opérer sur des capteurs hétérogènes. Toutefois, les données géospatiales ne sont pas nécessairement issues d'instruments de mesure. En particulier, les bases de données sémantiques, qu'elles soient institutionnelles ou commerciales, renferment une information géographique à un haut niveau d'abstraction qu'il est souhaitable de pouvoir prendre en compte.

5.3.1 OpenStreetMap

OpenStreetMap (OSM) est un SIG libre et participatif alimenté par un ensemble de contributeurs bénévoles qui cartographient les lieux qui leur sont familiers. Les contributeurs peuvent utiliser l'éditeur en ligne pour annoter des fonds de carte géoréférencés en y ajoutant ou en mettant à jour les éléments du réseau routier, les empreintes au sol des bâtiments, les espaces verts, etc. En plus de ces éditions manuelles, la communauté *OpenStreetMap* (OSM) utilise certaines sources de données officielles, comme le cadastre de l'État français afin de mettre à jour les limites administratives des entités géographiques ou pour importer les nouveaux bâtis. OSM est la plus grande base de données d'information géographique sous licence libre au monde, compilant une large taxonomie d'entités géographiques allant des autoroutes aux parcs de loisirs en passant par les églises, les cimetières et les parcelles agricoles.

Peu de travaux se sont penchés sur l'intégration des données OSM pour l'apprentissage automatique depuis l'ouverture du site en 2004. Le plus souvent, OSM est utilisé comme vérité terrain pour la détection de routes et de bâtiments [29, 25] dans un contexte d'apprentissage supervisé ou pour le recalage automatique d'images satellites [43]. Isola et al. [17] se sont intéressés à la génération automatique de tuiles OSM à partir d'images satellites, mais seulement à des fins visuelles, sans évaluation selon des métriques de classification. Pourtant, OSM est une source de données riche permettant d'extraire de l'information géographique sémantique de haut niveau. Chen et Zipf [6] ont de fait utilisé des méthodes d'apprentissage actif pour détecter automatiquement les objets non encore présents dans OSM afin de les suggérer aux contributeurs. Danylo et al. [8] ont utilisé des forêts aléatoires appliquées sur certaines couches de données OSM pour prédire des secteurs climatologiques locaux, tandis que Geiss et al. [10] se sont penchés sur la détection automatique de zones sujettes à des catastrophes naturelles.

Ici, nous suggérons d'utiliser les couches sémantiques d'OSM comme entrée à un réseau de neurones profond pour la cartographie automatisée. L'idée est d'exploiter la donnée sémantique, possiblement bruitée et incomplète d'OSM afin d'extraire de l'information plus riche à une résolution supérieure. En effet, cette approche est nouvelle et dépasse le cadre habituel de transformation image → OSM. À l'inverse, il s'agit d'exploiter et d'enrichir les multiples sources d'information existantes, qu'elles soient sous forme d'images ou de SIG.

Pour ce faire, nous considérons le jeu de données ISPRS Potsdam sur lequel nous récupérons les données OSM de 2017. Nous sélectionnons les couches correspondant aux routes, aux empreintes de bâtiments, aux zones d'eau et aux espaces verts. Les routes sont définies dans OSM comme une collection d'éléments linéaires. Durant la rasterisation, nous assignons à celles-ci une largeur arbitraire en fonction de leur type (environ 3,5 m par voie en ville). En outre, les bâtiments, espaces verts et zones d'eau ne correspondent pas nécessairement entièrement aux images aériennes du jeu de données, soit à cause de constructions nouvelles (les images ont été acquises en 2014), soit car les annotations OSM sont incomplètes. Nous générons ainsi un raster à la même résolution que les images RVB contenant 4 canaux binaires : un pour le masque des routes, un pour les bâtiments, un pour l'eau et un pour les espaces verts.

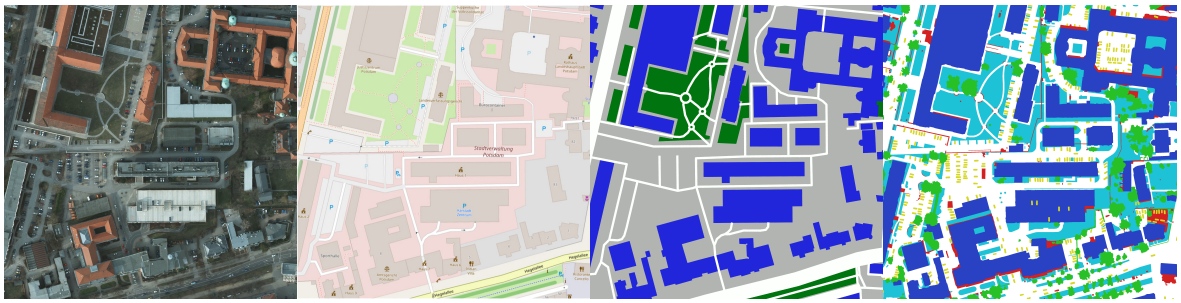


Image RVB | Tuile OSM | Raster OSM | Vérité terrain

FIGURE 5.9 – Tuile 4_12 du jeu de données ISPRS Potsdam et données OSM correspondantes. Légende : blanc : routes, bleu : bâtiments, cyan : végétation basse, vert : arbres, jaune : véhicules, rouge : autre.

5.3.2 Information *a priori* comme capteur virtuel

L'idée de notre approche est de traiter les données OSM comme provenant d'un capteur virtuel, c'est-à-dire comme une source de données complémentaire aux images optiques. Le raster ainsi formé contient une information partielle et incomplète, mais pouvant néanmoins faciliter le travail du réseau profond. En effet, plutôt que d'apprendre à reconnaître un bâtiment uniquement à partir de l'image optique, un réseau muni des deux sources de données peut se reposer en partie sur les annotations OSM pour repérer les bâtiments, et consacrer une partie de ses poids à gérer d'autres cas de figure plus difficiles. Plutôt que de réinventer entièrement la carte, le modèle peut donc incrémentalement venir enrichir l'information géographique préexistante à partir de la donnée optique, en contraste avec les approches traditionnelles. Nous appliquons donc les méthodes de fusion de données FuseNet et de correction résiduelle en utilisant comme entrées les images optiques et les couches OSM.

Lorsque les classes d'intérêt de la segmentation sémantique sont déjà présentes dans les données OSM, comme pour les bâtiments ou les routes, il est possible de les utiliser comme premières approximations de la vérité terrain. Celles-ci pourront ensuite être raffinées pour corriger les imprécisions de OSM et prédire les classes manquantes. Ce procédé s'apparente ainsi à l'apprentissage par résidu [14] et à l'apprentissage par raffinement [23], tous deux connus pour améliorer les performances des CNN et FCN.

Ici, nous utilisons un simple FCN composé du premier bloc convolutif de VGG-16 [40], afin de convertir les données raster OSM en cartes sémantiques approchant la vérité terrain. Ce modèle sera noté OSMNet par la suite. Les données optiques sont traitées par un FCN dérivé du modèle SegNet [4] en suivant l'approche du Chapitre 3. En appliquant ces deux modèles, nous pouvons alors calculer une carte de prédiction moyenne combinant les deux sources d'entrée. Dans ce cas, en notant I l'image couleur d'entrée, $\mathcal{O}$ le raster OSM, $\hat{\mathrm{P}}_{image}$ la fonction de prédiction de SegNet et $\hat{\mathrm{P}}_{OSM}$ la fonction de prédiction de OSMNet, alors la fonction de prédiction moyenne $\hat{\mathrm{P}}$ s'écrit :

$$\hat{\mathrm{P}}(\mathrm{I},\mathcal{O}) = \frac{1}{\alpha+\beta}(\alpha \cdot \hat{\mathrm{P}}_{image}(\mathrm{I}) + \beta \cdot \hat{\mathrm{P}}_{OSM}(\mathcal{O})) \,. \tag{5.7}$$

OSM contenant déjà une part conséquente de l'information géographique attendue, on peut supposer que $\hat{\mathrm{P}}_{OSM}(\mathcal{O})$ est une bonne approximation de la vérité terrain $\mathrm{P}_{réelle}$. $\hat{\mathrm{P}}_{image}$ s'écrit donc sous la forme d'un raffinement [23] :

$$\left\| \hat{\mathrm{P}}_{image}(\mathrm{I}) \right\| \propto \left\| \mathrm{P}_{réelle} - \hat{\mathrm{P}}_{OSM}(\mathcal{O}) \right\| \ll \left\| \mathrm{P}_{réelle} \right\| \,. \tag{5.8}$$

En outre, ceci peut encore s'exprimer sous la forme d'une correction résiduelle en introduisant le module éponyme C. En effet, en vertu de l'Équation (5.4), la prédiction $\hat{\mathrm{P}}^*$ après

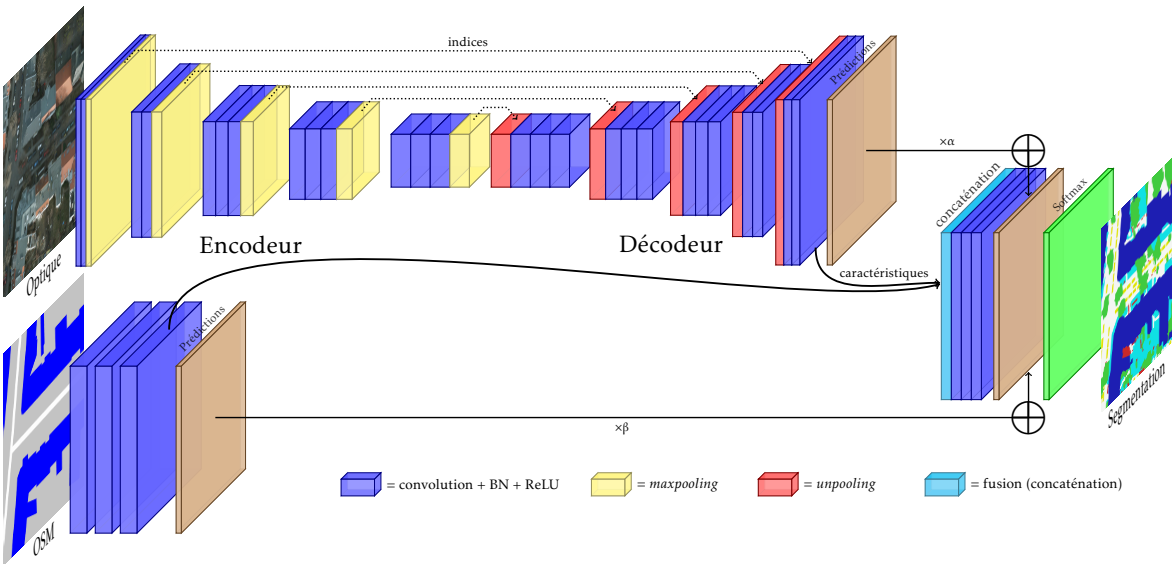


Figure 5.10 – Correction résiduelle appliquée à un OSMNet et un SegNet.

correction résiduelle s'écrit :

$$\hat{\mathrm{P}}^{*}(\mathrm{I},\mathcal{O}) = \hat{\mathrm{P}}(\mathrm{I},\mathcal{O}) + \mathrm{C}\left(\mathrm{Z}_{image}, \mathrm{Z}_{OSM}\right), \tag{5.9}$$

où Z_{image} et Z_{OSM} sont les cartes d'activation finales respectives de SegNet et OSMNet.

Dans ce cadre, l'apprentissage par résidu peut être conçu comme la modélisation d'un terme de correction d'erreur, illustré dans la Figure 5.10. Ainsi, le raffinement de la carte OSM de départ est lui-même corrigé par un résidu, selon un processus itératif à deux étapes.

De façon similaire, il est possible d'appliquer une architecture FuseNet sur I et $\mathcal{O}$, c'est-à-dire sur l'image couleur et le capteur virtuel OSM. Cela nécessite toutefois que les encodeurs de SegNet et d'OSMNet soient topologiquement identiques afin d'assurer la compatibilité des cartes d'activations au moment de la fusion, comme détaillé dans la Section 5.2.

5.3.3 Architectures multimodales pour l'information géographique

Nous utilisons le jeu de données ISPRS Potsdam sur lequel nous récupérons les données OSM correspondantes (cf. Figure 5.9). Les tuiles étant géo-référencées, nous générons les images OSM associées contenant les empreintes de routes, de bâtiments, des zones de végétation et de l'eau en utilisant Maperitive [1]. Les résultats sont obtenus par validation croisée sur 3 partitions du jeu de données.

Validation expérimentale

Nous considérons une nouvelle fois les hyperparamètres du Chapitre 3. Les résultats des modèles sont comparés avec ceux obtenus par forêt aléatoire (RF) sur l'image présegmentée en superpixels. Les caractéristiques utilisées sont les histogrammes de gradients orientés et de couleurs comme descripteur optique, ainsi que l'histogramme des classes OSM.

Les résultats obtenus sur les données de validation de l'ISPRS Potsdam sont détaillés dans le Tableau 5.4. Nous calculons les métriques définies dans la Section 3.3, c'est-à-dire le pourcentage global de pixels correctement classés et les scores F_1 pour chaque classe, sur la vérité terrain aux bordures érodées.

Comme attendu, l'inclusion de données OSM améliore les performances de classification du modèle, notamment pour les routes et les bâtiments qui bénéficient de l'information

1. http://maperitive.net/

Tableau 5.4 – Résultats de segmentation sémantique multimodale avec OSM sur le jeu de données ISPRS Potsdam (scores F_1 par classe et pourcentage global de pixels bien classés).

Méthode	Routes	Bâtiments	Vég. basse	Arbres	Véhicules	Global
RF IRRVB	77,0%	79,7%	73,1%	59,4%	58,8%	74,2%
SegNet RVB	93,0%	92,9%	85,0%	85,1%	95,1%	89,7%
RF IRRVB+OSM	85,6%	92,4%	73,8%	59,5%	67,6%	80,9%
CR RVB+OSM	93,9%	92,8%	85,1%	**85,2**%	95,8%	90,6%
FuseNet	**95,3**%	**95,9**%	**86,3**%	85,1%	**96,8**%	**92,3**%

géographique. En effet, cette information additionnelle permet de supprimer certaines ambiguïtés où un modèle purement optique aurait des difficultés, par exemple pour distinguer un parking au sol et sur un toit, à l'apparence très similaire. Il est par ailleurs intéressant de constater que des classes absentes des couches OSM bénéficient de cet apport en information, notamment les différents types de végétation et les véhicules.

Par ailleurs, l'intégration des données OSM dans l'apprentissage permet d'accélérer la convergence du modèle. Sur le même jeu de données, le modèle SegNet appris par raffinement depuis OSM nécessite 25% d'itérations en moins que le SegNet RVB classique et converge vers un meilleur optimal local, avec une fonction de coût à 0,39 contre 0,45, pour un même taux de réussite. Enfin, l'inclusion des données OSM rend la sortie du réseau visuellement plus cohérente et mieux structurée spatialement, comme illustré dans la Figure 5.12.

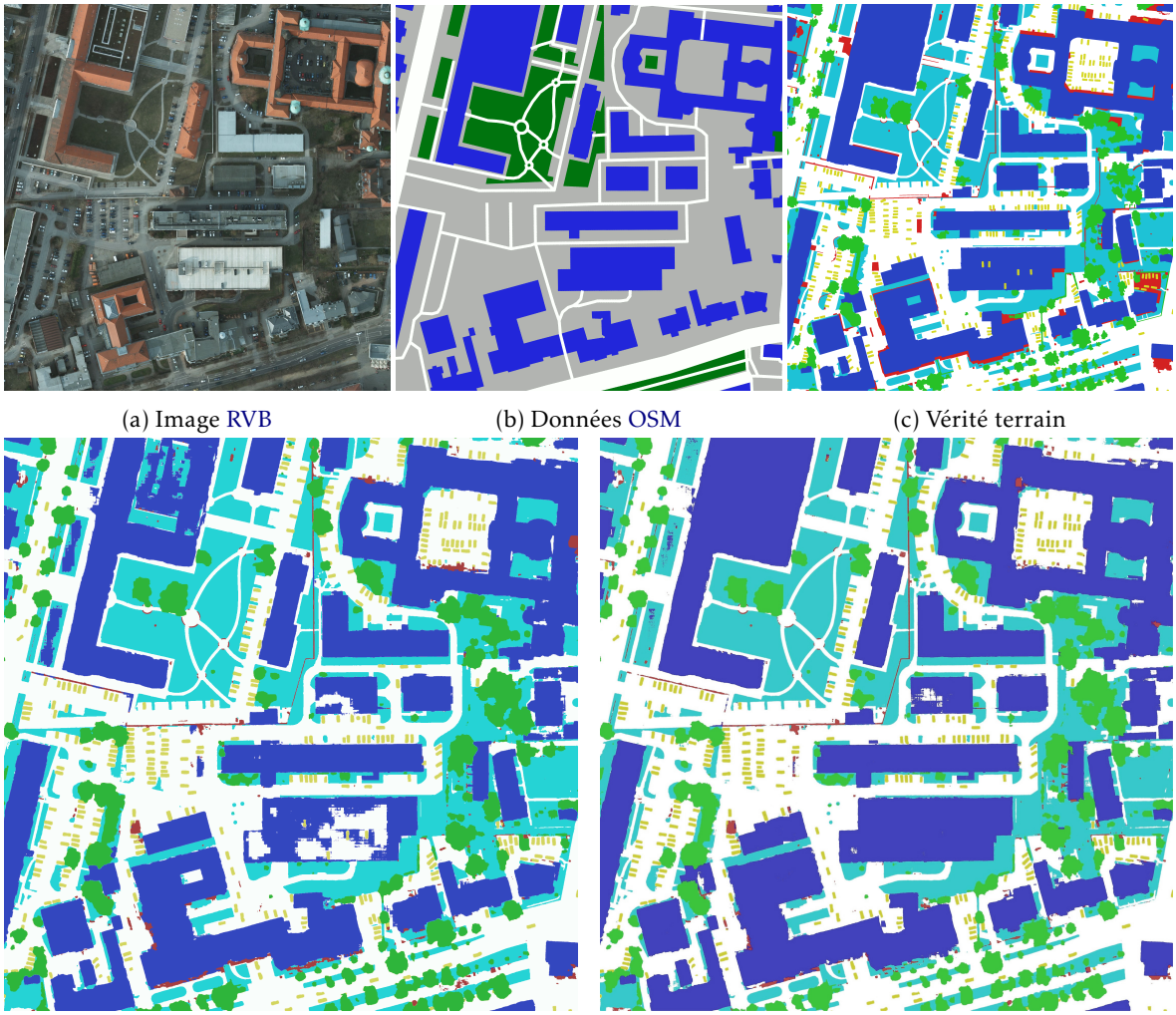
Dans l'ensemble, il apparaît que les FCN sont naturellement bien adaptés à l'apprentissage multimodal. En particulier, nous avons montré dans ce chapitre qu'il est possible de tirer profit des multiples sources de données géographiques hétérogènes, aussi bien issues de capteurs que de SIG. Les informations multisources permettent d'enrichir les capacités d'inférence de nos modèles aussi bien quantitativement que qualitativement sur les deux jeux de données que nous avons considérés. Enfin, les approches d'apprentissage multimodal et de fusion de prédictions permettent de faire face à différents types d'obstacles mais semblent bénéfiques dans tous les cas.

Les travaux présentés dans la Section 5.1 ont été le sujet d'un article de journal :

— Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « Beyond RGB : Very High Resolution Urban Remote Sensing with Multimodal Deep Networks ». Dans : *ISPRS Journal of Photogrammetry and Remote Sensing* (23 nov. 2017). ISSN : 0924-2716. DOI : 10.1016/j.isprsjprs.2017.11.011

Les travaux présentés dans la Section 5.3 ont été le sujet d'une publication en conférence :

— Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « Joint Learning from Earth Observation and OpenStreetMap Data to Get Faster Better Semantic Maps ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Honolulu, United States, juil. 2017, p. 1552-1560. DOI : 10.1109/CVPRW.2017.199



(a) Image RVB (b) Données OSM (c) Vérité terrain

(d) Prédiction (SegNet RVB) (e) Prédiction (FuseNet RVB+OSM)

Figure 5.11 – Exemple de segmentation obtenue sur le jeu de données ISPRS Potsdam en incluant la donnée OSM. Les erreurs sur les bâtiments sont grandement réduites.
Légende : blanc : routes, bleu : bâtiments, cyan : végétation basse, vert : arbres, jaune : véhicules, rouge : autre.

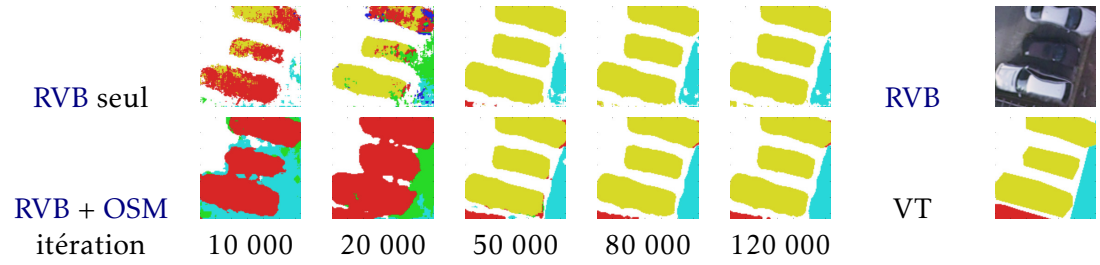


Figure 5.12 – Évolution des prédictions de SegNet RVB et RVB+OSM.
L'ajout de OSM rend les prédictions visuellement plus structurées.
Légende : blanc : routes, bleu : bâtiments, cyan : végétation basse, vert : arbres, jaune : véhicules, rouge : autre.

Références

[1] Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « Semantic Segmentation of Earth Observation Data Using Multimodal and Multi-Scale Deep Networks ». Dans : *Computer Vision – ACCV 2016*. Springer, Cham, 20 nov. 2016, p. 180-196. doi : 10.1007/978-3-319-54181-5_12 (cf. p. 124).

[2] Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « Beyond RGB : Very High Resolution Urban Remote Sensing with Multimodal Deep Networks ». Dans : *ISPRS Journal of Photogrammetry and Remote Sensing* (23 nov. 2017). issn : 0924-2716. doi : 10.1016/j.isprsjprs.2017.11.011 (cf. p. 134).

[3] Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « Joint Learning from Earth Observation and OpenStreetMap Data to Get Faster Better Semantic Maps ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Honolulu, United States, juil. 2017, p. 1552-1560. doi : 10.1109/CVPRW.2017.199 (cf. p. 134).

[4] Vijay Badrinarayanan, Alex Kendall et Roberto Cipolla. « SegNet : A Deep Convolutional Encoder-Decoder Architecture for Scene Segmentation ». Dans : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.12 (déc. 2017), p. 2481-2495. issn : 0162-8828. doi : 10.1109/TPAMI.2016.2644615 (cf. p. 121, 132).

[5] Tadas Baltrušaitis, Chaitanya Ahuja et Louis-Philippe Morency. « Multimodal Machine Learning : A Survey and Taxonomy ». Dans : (25 mai 2017). arXiv : 1705.09406 [cs]. url : http://arxiv.org/abs/1705.09406 (cf. p. 120, 124).

[6] Jiaoyan Chen et Alexander Zipf. « DeepVGI : Deep Learning with Volunteered Geographic Information ». Dans : *26th International World Wide Web Conference (Poster)*. ACM, 2017 (cf. p. 131).

[7] Camille Couprie et al. « Toward real-time indoor semantic segmentation using depth information ». Dans : *Journal of Machine Learning Research* (2014). issn : 1532-4435. url : https://nyuscholars.nyu.edu/en/publications/toward-real-time-indoor-semantic-segmentation-using-depth-informa (cf. p. 121).

[8] Olha Danylo et al. « Contributing to WUDAPT : A Local Climate Zone Classification of Two Cities in Ukraine ». Dans : *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 9.5 (mai 2016), p. 1841-1853. issn : 1939-1404, 2151-1535. doi : 10.1109/JSTARS.2016.2539977. url : http://ieeexplore.ieee.org/document/7447735/ (cf. p. 131).

[9] Andreas Eitel et al. « Multimodal Deep Learning for Robust RGB-D Object Recognition ». Dans : *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Sept. 2015, p. 681-687. doi : 10.1109/IROS.2015.7353446 (cf. p. 120, 121).

[10] Christian Geiss et al. « Joint Use of Remote Sensing Data and Volunteered Geographic Information for Exposure Estimation : Evidence from Valparaíso, Chile ». Dans : *Natural Hazards* 86.1 (1er mar. 2017), p. 81-105. issn : 0921-030X, 1573-0840. doi : 10.1007/s11069-016-2663-8 (cf. p. 131).

[11] Joris Guerry, Bertrand Le Saux et David Filliat. « "Look at This One" Detection Sharing between Modality-Independent Classifiers for Robotic Discovery of People ». Dans : *2017 European Conference on Mobile Robots (ECMR)*. 2017 European Conference on Mobile Robots (ECMR). Sept. 2017, p. 1-6. doi : 10.1109/ECMR.2017.8098679 (cf. p. 122).

[12] Hengkai Guo, Guijin Wang et Xinghao Chen. « Two-Stream Convolutional Neural Network for Accurate RGB-D Fingertip Detection Using Depth and Edge Information ». Dans : *Proceedings of the IEEE International Conference on Image Processing (ICIP)*. 2016, p. 2608-2612. doi : 10.1109/ICIP.2016.7532831 (cf. p. 121).

[13] Caner Hazirbas et al. « FuseNet : Incorporating Depth into Semantic Segmentation via Fusion-Based CNN Architecture ». Dans : *Computer Vision – ACCV 2016*. Asian Conference on Computer Vision. Springer, Cham, 20 nov. 2016, p. 213-228. DOI : 10.1007/978-3-319-54181-5_14 (cf. p. 121-123).

[14] Kaiming He et al. « Deep Residual Learning for Image Recognition ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, United States, juin 2016, p. 770-778. DOI : 10.1109/CVPR.2016.90 (cf. p. 125, 126, 132).

[15] Micah Hodosh, Peter Young et Julia Hockenmaier. « Framing Image Description As a Ranking Task : Data, Models and Evaluation Metrics ». Dans : *Journal of Artificial Intelligence Research* 47.1 (mai 2013), p. 853-899. ISSN : 1076-9757. URL : http://dl.acm.org/citation.cfm?id=2566972.2566993 (cf. p. 121).

[16] Judy Hoffman, Saurabh Gupta et Trevor Darrell. « Learning with Side Information through Modality Hallucination ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, p. 826-834. URL : http://www.cv-foundation.org/openaccess/content_cvpr_2016/html/Hoffman_Learning_With_Side_CVPR_2016_paper.html (cf. p. 128).

[17] Phillip Isola et al. « Image-to-Image Translation with Conditional Adversarial Networks ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, United States, juil. 2017, p. 5967-5976. DOI : 10.1109/CVPR.2017.632 (cf. p. 131).

[18] Michael Kampffmeyer, Arnt-Borre Salberg et Robert Jenssen. « Semantic Segmentation of Small Objects and Modeling of Uncertainty in Urban Remote Sensing Images Using Deep Convolutional Neural Networks ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Las Vegas, United States, 2016, p. 1-9 (cf. p. 128).

[19] Andrej Karpathy et Li Fei-Fei. « Deep Visual-Semantic Alignments for Generating Image Descriptions ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, p. 3128-3137 (cf. p. 120).

[20] Yelin Kim, Honglak Lee et Emily Mower Provost. « Deep Learning for Robust Feature Generation in Audiovisual Emotion Recognition ». Dans : *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. Mai 2013, p. 3687-3691. DOI : 10.1109/ICASSP.2013.6638346 (cf. p. 120).

[21] Adrien Lagrange et al. « Benchmarking Classification of Earth-Observation Data : From Learning Explicit Features to Convolutional Networks ». Dans : *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS). Juil. 2015, p. 4173-4176. DOI : 10.1109/IGARSS.2015.7326745 (cf. p. 121, 122).

[22] Seungyong Lee, Seong-Jin Park et Ki-Sang Hong. « RDFNet : RGB-D Multi-Level Residual Feature Fusion for Indoor Semantic Segmentation ». Dans : *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. Oct. 2017, p. 4990-4999. DOI : 10.1109/ICCV.2017.533 (cf. p. 122).

[23] Guosheng Lin et al. « RefineNet : Multi-Path Refinement Networks with Identity Mappings for High-Resolution Semantic Segmentation ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Juil. 2017, p. 5168-5177. DOI : 10.1109/CVPR.2017.549 (cf. p. 132).

[24] Yansong Liu et al. « Dense Semantic Labeling of Very-High-Resolution Aerial Imagery and LiDAR with Fully-Convolutional Neural Networks and Higher-Order CRFs ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Honolulu, United States, juil. 2017, p. 1561-1570. DOI : 10.1109/CVPRW.2017.200 (cf. p. 122, 127).

[25] Emmanuel Maggiori. « Learning Approaches for Large-Scale Remote Sensing Image Classification ». Thèse de doct. Université Côte d'Azur, 22 juin 2017. URL : https://hal.inria.fr/tel-01589661/document (cf. p. 131).

[26] Dimitrios Marmanis et al. « Classification With an Edge : Improving Semantic Image Segmentation with Boundary Detection ». Dans : *ISPRS Journal of Photogrammetry and Remote Sensing* (2017). DOI : 10.1016/j.isprsjprs.2017.11.009. arXiv : 1612.01337 (cf. p. 127, 128).

[27] Uwe Meier, Wolfgang Hürst et Paul Duchnowski. « Adaptive Bimodal Sensor Fusion for Automatic Speechreading ». Dans : *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*. 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings. T. 2. Mai 1996, 833-836 vol. 2. DOI : 10.1109/ICASSP.1996.543250 (cf. p. 121).

[28] Bjoern H. Menze et al. « The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS) ». Dans : *IEEE Transactions on Medical Imaging* 34.10 (oct. 2015), p. 1993-2024. ISSN : 0278-0062. DOI : 10.1109/TMI.2014.2377694 (cf. p. 121).

[29] Volodymyr Mnih. « Machine Learning for Aerial Image Labeling ». University of Toronto, 2013 (cf. p. 131).

[30] Jiquan Ngiam et al. « Multimodal Deep Learning ». Dans : *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*. 2011, p. 689-696. URL : http://machinelearning.wustl.edu/mlpapers/paper_files/ICML2011Ngiam_399.pdf (cf. p. 120, 121).

[31] Kuniaki Noda et al. « Audio-Visual Speech Recognition Using Deep Learning ». Dans : *Applied Intelligence* 42.4 (1er juin 2015), p. 722-737. ISSN : 0924-669X, 1573-7497. DOI : 10.1007/s10489-014-0629-7 (cf. p. 121).

[32] Ferda Ofli et al. « Berkeley MHAD : A Comprehensive Multimodal Human Action Database ». Dans : *2013 IEEE Workshop on Applications of Computer Vision (WACV)*. Jan. 2013, p. 53-60. DOI : 10.1109/WACV.2013.6474999 (cf. p. 121).

[33] Francisco Javier Ordóñez et Daniel Roggen. « Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition ». Dans : *Sensors* 16.1 (18 jan. 2016), p. 115. DOI : 10.3390/s16010115. URL : http://www.mdpi.com/1424-8220/16/1/115 (cf. p. 120).

[34] Sakrapee Paisitkriangkrai et al. « Effective Semantic Pixel Labelling with Convolutional Networks and Conditional Random Fields ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Juin 2015, p. 36-43. DOI : 10.1109/CVPRW.2015.7301381 (cf. p. 122).

[35] Fabien Ringeval et al. « Introducing the RECOLA Multimodal Corpus of Remote Collaborative and Affective Interactions ». Dans : *10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*. Avr. 2013, p. 1-8. DOI : 10.1109/FG.2013.6553805 (cf. p. 121).

[36] Björn Schuller et al. « AVEC 2011–The First International Audio/Visual Emotion Challenge ». Dans : *Affective Computing and Intelligent Interaction*. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, 2011, p. 415-424. ISBN : 978-3-642-24570-1 978-3-642-24571-8. DOI : 10.1007/978-3-642-24571-8_53 (cf. p. 121).

[37] Max Schwarz, Hannes Schulz et Sven Behnke. « RGB-D Object Recognition and Pose Estimation Based on Pre-Trained Convolutional Neural Network Features ». Dans : *2015 IEEE International Conference on Robotics and Automation (ICRA)*. Mai 2015, p. 1329-1335. doi : 10.1109/ICRA.2015.7139363 (cf. p. 121).

[38] Jamie Sherrah. « Fully Convolutional Networks for Dense Semantic Labelling of High-Resolution Aerial Imagery ». Dans : (8 juin 2016). arXiv : 1606.02585 [cs]. url : http://arxiv.org/abs/1606.02585 (cf. p. 127).

[39] Karen Simonyan et Andrew Zisserman. « Two-Stream Convolutional Networks for Action Recognition in Videos ». Dans : *Advances in Neural Information Processing Systems 27*. Curran Associates, Inc., 2014, p. 568-576. url : http://papers.nips.cc/paper/5353-two-stream-convolutional-networks-for-action-recognition-in-videos.pdf (cf. p. 121).

[40] Karen Simonyan et Andrew Zisserman. « Very Deep Convolutional Networks for Large-Scale Image Recognition ». Dans : *Proceedings of the International Conference on Learning Representations (ICLR)*. Mai 2015. url : http://arxiv.org/abs/1409.1556 (cf. p. 132).

[41] Xinhang Song, Shuqiang Jiang et Luis Herranz. « Combining Models from Multiple Sources for RGB-D Scene Recognition ». Dans : *Proceedings of the 26th International Joint Conference on Artificial Intelligence*. IJCAI'17. Melbourne, Australia : AAAI Press, 2017, p. 4523-4529. isbn : 978-0-9992411-0-3 (cf. p. 121).

[42] Nitish Srivastava et Ruslan Salakhutdinov. « Multimodal Learning with Deep Boltzmann Machines ». Dans : *Journal of Machine Learning Research* 15 (2014), p. 2949-2980. issn : 1532-4435 (cf. p. 121).

[43] Maria Vakalopoulou et al. « Simultaneous Registration, Segmentation and Change Detection from Multisensor, Multitemporal Satellite Image Pairs. » Dans : International Geoscience and Remote Sensing Symposium (IGARSS). 10 juil. 2016. doi : 10.1109/IGARSS.2016.7729469 (cf. p. 131).

[44] Cihang Xie et al. « Adversarial Examples for Semantic Segmentation and Object Detection ». Dans : *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. Oct. 2017, p. 1378-1387. isbn : 978-1-5386-1032-9. doi : 10.1109/ICCV.2017.153 (cf. p. 128).

[45] Ben P. Yuhas, Moise H. Goldstein et Terrence J. Sejnowski. « Integration of Acoustic and Visual Speech Signals Using Neural Networks ». Dans : *IEEE Communications Magazine* 27.11 (1989), p. 65-71 (cf. p. 121).

Chapitre

6 Généralisabilité des modèles

I see no limit to the capabilities of machines. As microchips get smaller and faster, I can see them getting better than we are. I can visualize a time in the future when we will be to robots as dogs are to humans.

— Claude Shannon

Sommaire

6.1 Génération de données synthétiques 142
6.1.1 Modèles génératifs adversaires 143
6.1.2 Cadre expérimental 144
6.1.3 Analyse des spectres générés 145
6.1.4 Augmentation de données 147
6.2 Cas des données massives 149
6.2.1 Diversité des scènes 149
6.2.2 MiniFrance 151

Résumé du chapitre :

La généralisation des modèles entraînés à partir d'une vérité terrain aux nouvelles acquisitions est la clé pour une application à large échelle de l'apprentissage profond en observation de la Terre. En effet, la variabilité des environnements observés, aussi bien dans l'espace que dans le temps, limite la portée de modèles entraînés sur une poignée de scènes locales. Deux problèmes analogues se posent.

D'une part, nous savons que la création d'une vérité terrain est particulièrement coûteuse et complexe pour certains capteurs, notamment hyperspectraux. Or, l'entraînement de modèles statistiques sur des petits jeux de données encourage le surapprentissage et va à l'encontre des besoins en généralisabilité. Par conséquent, nous nous intéressons tout d'abord aux possibilités de génération de données d'apprentissage synthétiques afin de pallier l'absence d'exemples réels.

D'autre part, l'abondance de données de télédétection et surtout leur diversité soulève des interrogations quant à la capacité des réseaux profonds introduits jusqu'ici à passer à l'échelle. Une version simplifiée du problème consiste à étudier les possibilités de transfert de connaissances d'une scène à une autre pour un même modèle. Nous expérimentons ainsi avec les jeux de données ISPRS Potsdam et Vaihingen afin d'évaluer comment les modèles préentraînés peuvent être appliqués sur de nouvelles acquisitions pas ou peu annotées. Enfin, nous introduisons le jeu de données annoté à large échelle *MiniFrance*, le plus grand disponible publiquement à notre connaissance, regroupant 16 agglomérations en France métropolitaine et des annotations d'occupation des sols et d'empreintes de bâtiments.

6.1 Génération de données synthétiques

Comme nous l'avons vu dans le Chapitre 4, il existe relativement peu de jeux de données annotés en imagerie hyperspectrale et ceux disponibles sont de petite taille. La difficulté d'annotation mais aussi la faible résolution spatiale des capteurs rend en effet complexe la constitution de jeux de données massifs. Des bases de spectres mesurés en laboratoire, comme celle de l'*United States Geological Survey* (USGS) [1], sont en pratique difficilement exploitables compte-tenu des différences de capteurs, de calibrations et de conditions d'acquisition. Il semble de ce fait pertinent de s'interroger sur les possibilités offertes par l'augmentation de données pour la classification de spectres.

L'augmentation de données consiste à introduire des échantillons synthétiques afin d'enrichir un jeu d'apprentissage [8]. Cette pratique est particulièrement courante pour l'entraînement des CNN depuis l'article séminal de Krizhevsky, Sutskever et Hinton [12] afin d'éviter le surapprentissage. Dans un contexte de classification de données hyperspectrales, la rareté des échantillons annotés rend l'augmentation de données d'autant plus attrayante. Cependant, la plupart des travaux de l'état de l'art appliquant des CNN 2D ou 3D à la classification de telles images [6, 17, 21, 13] se sont limités à des jeux de données de taille restreinte, ne permettant pas d'exploiter au mieux les capacités d'apprentissage de représentation offertes par les réseaux profonds.

Ainsi, quelques études se sont penchées sur l'enrichissement artificiel des jeux de données hyperspectraux mis à la disposition de la communauté. Windrim et al. [23] ont proposé un modèle physique permettant de simuler les déformations d'un spectre s'il était soumis à des conditions d'illumination différentes de celle de l'acquisition originale, ce qui leur permet d'introduire une invariance à ces changements environnementaux. Toutefois, cela nécessite la conception et la mise en œuvre d'un modèle physique expert qui n'est pas générique, introduisant une phase d'estimation de l'illumination pouvant être imprécise dans des images de télédétection. Plus simplement, Chen et al. [6] augmentent le nombre d'échantillons disponibles en générant des combinaisons linéaires des spectres existants et en ajoutant du bruit gaussien, en supposant ces altérations plausibles. Enfin, Acquarelli et al. [1] proposent de propager les annotations d'un pixel à ses voisins par *clustering* afin d'incorporer des pixels observés mais initialement non annotés dans le jeu d'apprentissage. Si cette approche permet en effet d'apprendre à partir de pixels supplémentaires, le nombre total d'échantillons est toutefois borné par la taille de l'image acquise.

Ainsi, nous introduisons la problématique suivante : comment enrichir les bases d'apprentissage lorsque aucun *a priori* physique n'est disponible, en ajoutant autant de nouveaux échantillons que désiré ? Une première piste de réponse se trouve dans les travaux de Gemp et al. [9]. Ceux-ci implémentent des autoencodeurs variationnels qu'ils utilisent comme modèles génératifs pour le démélange de spectres, afin de déterminer les *endmembers* d'une image. Pour la classification, Davari et al. [7] entraînent un modèle de mélange gaussien, ou *Gaussian Mixture Model*, (GMM) afin d'estimer la distribution des caractéristiques spectrales après extraction des profils d'attributs. Ils peuvent ensuite générer de nouveaux vecteurs d'attributs synthétiques à partir de la distribution estimée, qui viennent enrichir le jeu d'apprentissage original.

Les modèles génératifs permettent en effet d'estimer les paramètres d'une distribution statistique latente à un ensemble d'observations pour en échantillonner de nouvelles. Nous proposons donc d'utiliser des modèles génératifs pour approcher la distribution latente des spectres au sein d'une image hyperspectrale afin de synthétiser de nouveaux échantillons susceptible d'appartenir à celle-ci. Il s'agit d'une approche ancrée uniquement dans les données, ne nécessitant aucun *a priori* physique ni modèle de capteur.

En particulier, nous proposons d'utiliser les réseaux génératifs adversaires, ou *Generative Adversarial Networks* (GAN) [10] afin d'estimer la distribution des spectres de l'image, puis

1. *USGS Spectral Library* : https://speclab.cr.usgs.gov/spectral-lib.html

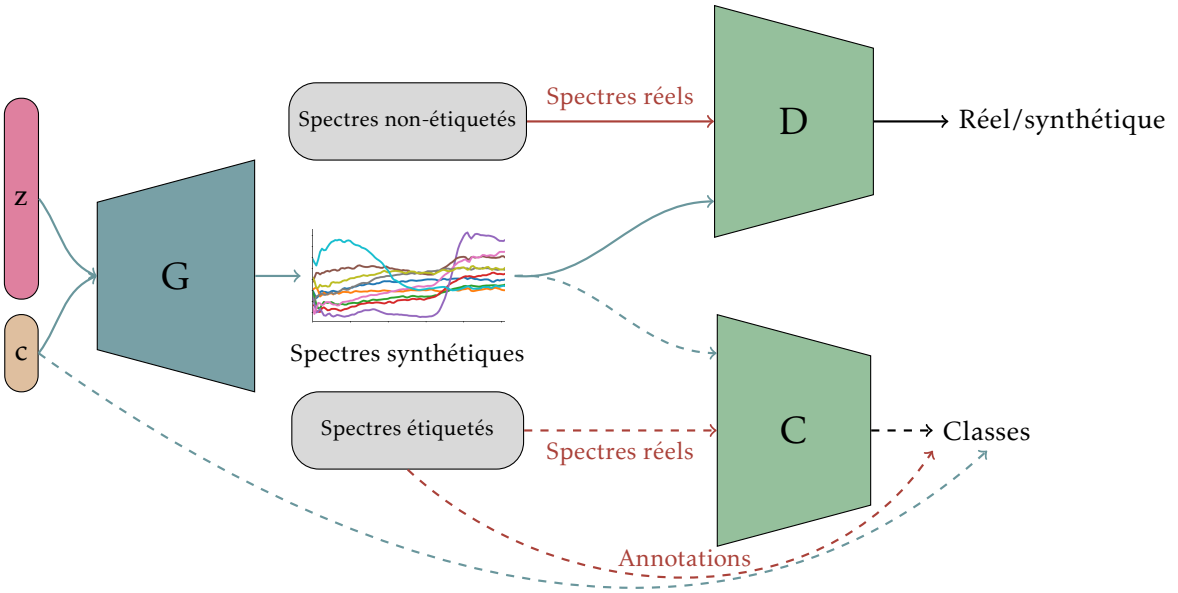


Figure 6.1 – La structure du GAN utilisé pour la synthèse de spectres artificiels. Les flèches en rouge indiquent l'entraînement du classifieur et du discriminateur, tandis que les flèches en bleu indiquent l'entraînement du générateur. Les flèches en pointillé indiquent les connexions qui ne sont utilisées que dans le cadre supervisé.

de générer de nouveaux spectres dont la présence dans la distribution réelle est statistiquement plausible. Cette méthode se veut semi-supervisée afin de bénéficier aussi bien de la connaissance des spectres annotés que des spectres non-annotés. Nous validons l'intérêt d'utiliser ces spectres artificiels pour l'augmentation de données sur plusieurs jeux de données hyperspectraux publics, aériens comme satellitaires, sur différentes zones géographiques.

6.1.1 Modèles génératifs adversaires

Le principe des GAN a été introduit par Goodfellow et al. [10] en 2014. L'idée est d'utiliser des réseaux de neurones profonds pour modéliser la distribution statistique sous-jacente à un ensemble d'observations. Un générateur est ainsi entraîné pour approcher la projection entre un espace latent de bruit gaussien vers la distribution empirique observée. Cependant, la distribution n'est observée que sur quelques échantillons et l'on souhaite utiliser le générateur pour créer de nouveaux échantillons probables. Pour ce faire, le générateur est entraîné pour approcher la distribution à l'aide d'une fonction objectif adversaire. Cette fonction est implicitement définie en introduisant un second réseau, appelé discriminateur ou critique. Le discriminateur apprend à estimer si un échantillon provient de l'ensemble des données réelles ou bien a été généré artificiellement. À chaque étape de l'optimisation, le discriminateur est entraîné sur quelques itérations afin de lui permettre d'estimer la frontière entre données réelles et données synthétiques. Le générateur est ensuite optimisé de telle sorte à ce qu'il *piège* le discriminateur, c'est-à-dire que les échantillons synthétisés soient indistinguables des exemples réels du point de vue du critique (cf. Définition 7).

Plusieurs variantes des GAN ont été proposées depuis leur introduction. Nous utilisons ici un générateur G et un discriminateur D sur le principe des Wasserstein GAN [2] avec la régularisation de Gulrajani et al. [11], dont l'entraînement est prévu pour minimiser la distance de Wasserstein entre la distribution réelle et la distribution synthétique. G transforme un vecteur de bruit aléatoire z en un spectre, de sorte que D estime que celui-ci appartient à la distribution réelle. Formellement, Goodfellow et al. [10] présentent les GAN

comme un algorithme minimax à deux joueurs ayant pour fonction de valuation V :

$$\min_{\mathrm{G}} \max_{\mathrm{D}} \mathrm{V}(\mathrm{G},\mathrm{D}) = \mathbb{E}_{x\sim\text{données},z\sim p_z}[\mathrm{D}(x) - \mathrm{D}(\mathrm{G}(z))] \ . \tag{6.1}$$

Toutefois, ce mode de fonctionnement est non-supervisé, c'est-à-dire qu'il n'est possible que de générer de nouveaux échantillons sans contrôle sur leur classe. Il serait possible de créer un générateur pour chaque classe, mais cela serait coûteux en temps et en mémoire. Dans notre cas, nous souhaitons pouvoir *conditionner* le générateur par rapport à la classe du spectre que nous souhaitons synthétiser. Nous utilisons ainsi un classifieur auxiliaire C [18] qui ajoute une contrainte supplémentaire lors de l'optimisation du générateur en s'assurant que les spectres générés sont bien classifiés dans la classe choisie. Dans ce cas, G considère également un vecteur de conditionnement c qui encode une classe sous forme *one-hot*. Les spectres générés par G doivent également être correctement classifiés par C. Le terme de classification additionnel s'écrit ainsi :

$$\mathrm{L}_c = \mathbb{E}_{x\sim\text{données}}[\log \mathrm{P}(\mathrm{C}(x) = c)] + \mathbb{E}_{z\sim p_z}[\log \mathrm{P}(\mathrm{C}(\mathrm{G}(z) = c)] \tag{6.2}$$

et sera ajouté avec un facteur λ_c permettant de contrôler l'importance accordée à la partie supervisée.

L'architecture complète est détaillée dans la Figure 6.1. Si G et D peuvent être entraînés sans annotation, c'est-à-dire de façon non-supervisée, C doit être entraîné sur des exemples étiquetés. L'ensemble est donc semi-supervisé et peut exploiter simultanément les échantillons annotés et non-annotés sur l'ensemble de l'hypercube.

Définition 7. *Algorithme d'entraînement de réseaux de neurones génératifs adversaires :*

Avec n la taille du batch, $\mathcal{Z}$ *la distribution latente,* Ω *l'ensemble des échantillons (annotés ou non),* $\Omega^* \subset \Omega$ *le sous-ensemble des éléments annotés et* $\mathcal{L}$ *l'entropie croisée. Tant que le critère de convergence n'est pas atteint :*

1. *Optimisation de* D. *Répéter* k_D *fois :*
   - *Tirer aléatoirement un vecteur* $\mathbf{x}$ *de n éléments dans* Ω
   - *Itérer la descente de gradient sur* D *pour maximiser* $\mathrm{D}(\mathbf{x})$
   - *Tirer aléatoirement un vecteur* $\mathbf{z}$ *de n éléments dans* $\mathcal{Z}$
   - *Itérer la descente de gradient sur* D *pour minimiser* $\mathrm{D}(\mathrm{G}(\mathbf{z}))$
2. *(facultatif) Optimisation de* C. *Répéter* k_C *fois :*
   - *Tirer aléatoirement un vecteur* $\mathbf{x}^*$ *de n éléments dans* Ω^**, avec y le vecteur de classes*
   - *Itérer la descente de gradient sur* C *pour minimiser* $\mathcal{L}(\mathrm{C}(\mathbf{x}^*), y)$
3. *Optimisation de* G.
   - *Tirer aléatoirement un vecteur* $\mathbf{z}$ *de n éléments dans* $\mathcal{Z}$
   - *(facultatif) Générer et concaténer à* $\mathbf{z}$ *son vecteur de condition* $\mathbf{c}$
   - *Générer les échantillons* $\hat{\mathbf{x}} = \mathrm{G}(\mathbf{z})$
   - *Calculer la fonction de coût* $\mathcal{L}_{totale}(\mathbf{z}) = -\mathrm{D}(\hat{\mathbf{x}})$
   - *(facultatif) Ajouter l'erreur sur* C *:* $\mathcal{L}_{totale}(\mathbf{z}) := \mathcal{L}_{totale}(\mathbf{z}) + \mathcal{L}(\mathrm{C}(\hat{\mathbf{x}}), \mathbf{c}))$
   - *Itérer la descente de gradient sur* G *pour minimiser* $\mathcal{L}_{totale}$

6.1.2 Cadre expérimental

Nous entraînons ce GAN sur les jeux de données Pavia University, Pavia Center, Indian Pines et Botswana (cf. Section 4.2.2) en utilisant les réflectances corrigées en atmosphère lorsqu'elles sont disponibles. Comme nous essayons de générer des spectres individuels et non des hypercubes, nous utilisons pour G, D et C des réseaux simples, entièrement connectés à 4 couches, utilisant la fonction d'activation *Leaky ReLU* [15]. L'intérêt d'une telle fonction d'activation par rapport à la ReLU usuelle est d'avoir un gradient non-nul sur toute sa plage de fonctionnement, ce qui facilite la rétropropagation du gradient de D vers C. La sortie de

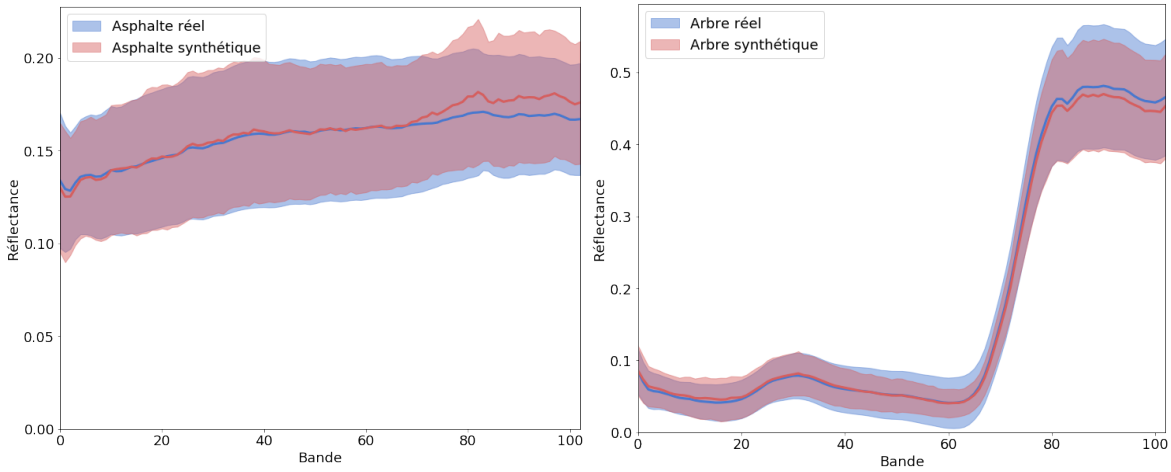


FIGURE 6.2 – Spectre moyen et écart-type pour deux classes de matériaux du jeu de données Pavia Center. Les échantillons synthétiques moyens sont plus bruités et sont surappris sur certaines propriétés spectrales locales.

G est suivie par une sigmoïde pour contraindre les valeurs de réflectance synthétisée entre 0 et 1. D ne possède qu'une seule sortie et C possède autant de sorties que le jeu de données a de classes.

L'optimisation des trois réseaux se fait en utilisant la politique de descente de gradient stochastique *RMSProp* [22]. L'ensemble est entraîné durant 100 000 itérations avec une taille de *batch* de 256, C et D étant entraînés 2 fois à chaque itération. Lors de l'optimisation de G, la fonction est de coût auxiliaire de classification est pondérée par un facteur 0,2. Le taux d'apprentissage global est fixé à 5×10^{-5}.

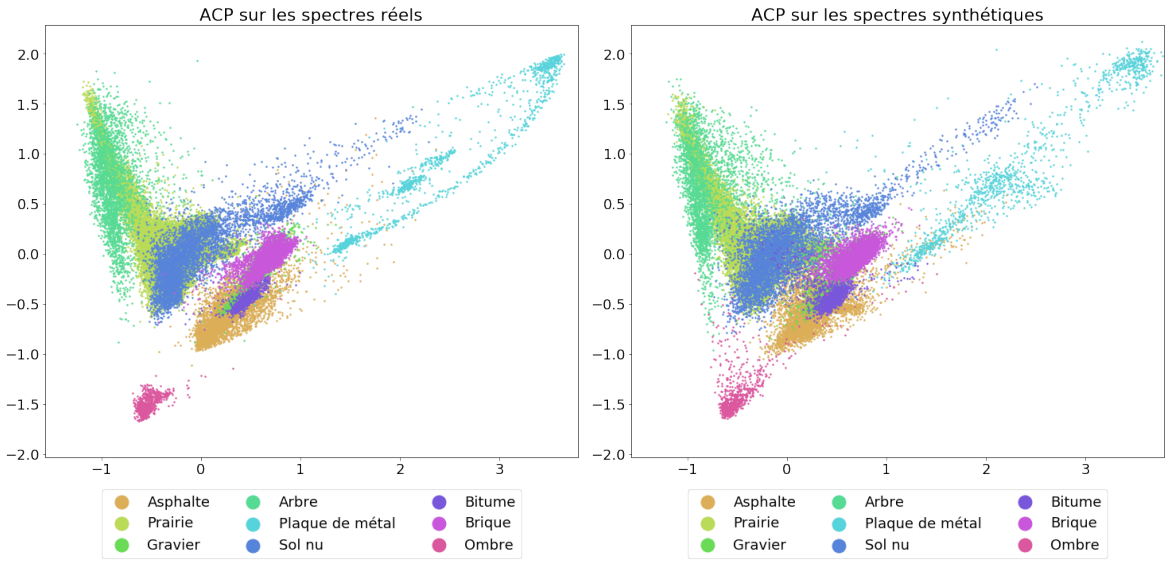
Par ailleurs, il apparaît nécessaire d'établir une performance de référence afin d'évaluer la pertinence des GAN pour la synthèse de spectres. Nous implémentons donc un modèle de mélange gaussien en utilisant la bibliothèque scikit-learn [19]. Nous reconstruisons un mélange pour chaque classe du jeu de données utilisant 10 composantes, que nous utilisons par la suite pour générer de nouveaux spectres.

6.1.3 Analyse des spectres générés

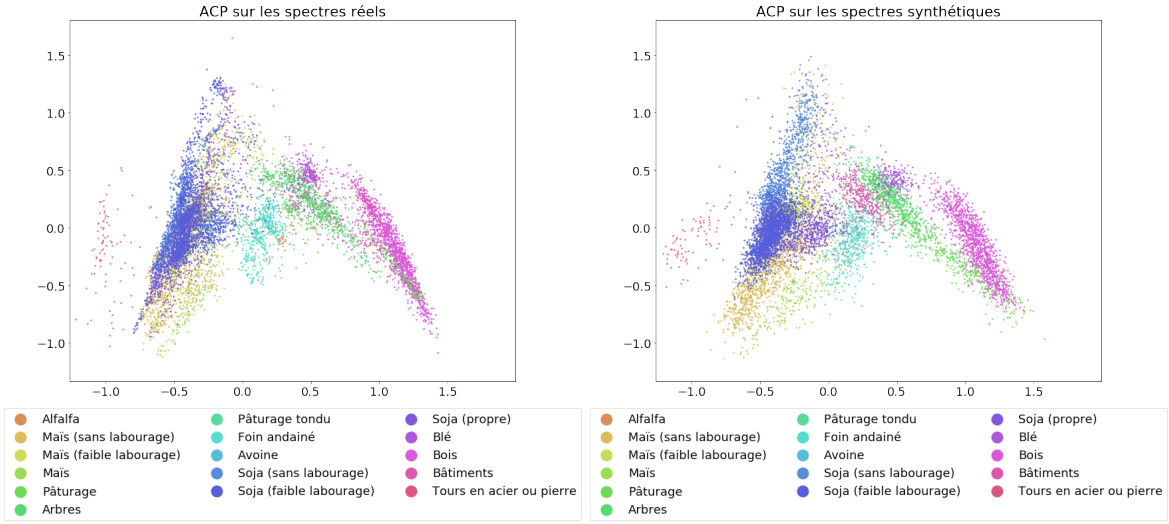
Dans un premier temps, nous cherchons à comparer selon plusieurs critères les distributions synthétiques et réelles. Pour ce faire, nous entraînons d'abord deux GAN sur Pavia University et Indian Pines.

Visuellement, il est possible de constater dans la Figure 6.2 que les spectres générés présentent des moments statistiques très similaires aux spectres réels. Les formes globales des spectres sont correctement approchées pour chaque classe. Toutefois, deux points négatifs sont identifiables. Tout d'abord, les spectres synthétiques moyens semblent plus bruités que leurs équivalents réels, ce qui signifie que le GAN a surappris certaines particularités liées aux échantillons d'entraînement choisis. En outre, l'écart-type de la distribution synthétique est inférieur à celui de la distribution réelle, ce qui signifie que les faux spectres sont moins diversifiés que les vrais. Ces deux constatations indiquent que le générateur souffre partiellement d'une forme d'apprentissage appelée *mode collapse* [20].

Pour mieux comprendre l'impact de ce surapprentissage, nous appliquons une ACP afin de projeter les spectres réels et synthétiques dans un espace de représentation à deux dimensions (cf. Figure 6.3). Les groupes formés par les différentes classes sont correctement reproduits par les échantillons synthétiques. Cependant, la distribution synthétique présente également quelques déformations montrant que, si le GAN a bien réussi à modéliser la forme générale des différents types de spectres, il n'est cependant pas parvenu à reproduire l'ensemble de leurs spécificités.



(a) Pavia University



(b) Indian Pines

Figure 6.3 – ACP sur les spectres réels et synthétiques. Les spectres réels correspondent à l'ensemble des échantillons annotés de l'image. Les deux ensembles contiennent le même nombre d'éléments.

Nous pouvons essayer d'estimer comment la distribution synthétique respecte les frontières entre classes de la distribution réelle en entraînant une SVM linéaire sur les spectres réels et en l'appliquant pour séparer les spectres synthétiques. La SVM va calculer les meilleurs hyperplans séparateurs pour la véritable distribution. Idéalement, ces hyperplans devraient séparer les spectres synthétiques exactement de la même façon. S'ils séparent nettement moins bien les spectres synthétiques, alors cela signifie que le générateur créé des échantillons irréalistes. S'ils séparent nettement mieux les échantillons, alors le générateur créé des exemples synthétiques trop similaires entre eux et regroupés autour des centroïdes correspondant aux classes réelles. Les résultats sont détaillés dans le Tableau 6.1. Nous considérons deux approches : entraînement sur 3% des spectres tirés au hasard uniformément ou sur 50% de l'image, disjoint spatialement de la zone de validation. Dans le mode non-supervisé, nous utilisons également les échantillons non-annotés. Comme attendu, la SVM sépare plus facilement les échantillons synthétiques que les spectres réels. Toutefois, entraîner la SVM sur les spectres synthétiques uniquement permet tout de même de séparer les spectres réels dans une certaine mesure. Autrement dit, si les échantillons synthétiques sont moins diversifiés que les spectres réels, ils sont néanmoins représentatifs des différentes

Tableau 6.1 – Exactitudes d'une SVM linéaire appliquée sur les spectres réels et synthétiques (Pavia University).

Découpage	Aléatoire (uniforme) – 3% (r)		Disjoint – 50% (s)	
Apprentissage \ Test	Réels	Synthétiques	Réels	Synthétiques
Réels	89,5	98,3	87,2	98,8
Synthétiques	87,8	99,2	79,4	99,9

classes, et ce alors même que ces spectres sont générés *ex nihilo* à partir de bruit aléatoire.

Finalement, comme les GAN permettent d'établir une projection entre un espace de représentation latent et le signal, il est possible d'explorer la variété des spectres en interpolant de façon continue entre deux points de l'espace latent. En effet, si z_1 et z_2 sont deux vecteurs aléatoires tirés dans la distribution gaussienne latente, alors il est possible d'interpoler entre les deux le long de l'hypersphère unité :

$$\begin{cases} \forall \alpha \in [0,1], \;\; z_\alpha = \frac{\sin((1-\alpha)\cdot\omega)}{\sin\omega} \cdot z_1 + \frac{\sin(\alpha\cdot\omega)}{\sin\omega} \cdot z_2 \\ \hat{x}_\alpha = \mathrm{G}(z_\alpha, c) \text{ avec } x_0 = \mathrm{G}(z_1, c) \text{ et } x_1 = \mathrm{G}(z_2, c) \end{cases} \tag{6.3}$$

avec ω l'angle entre z_1 et z_2. De la même façon, il est possible d'interpoler à vecteur de bruit fixe entre deux vecteurs de conditionnement c_1 et c_2 :

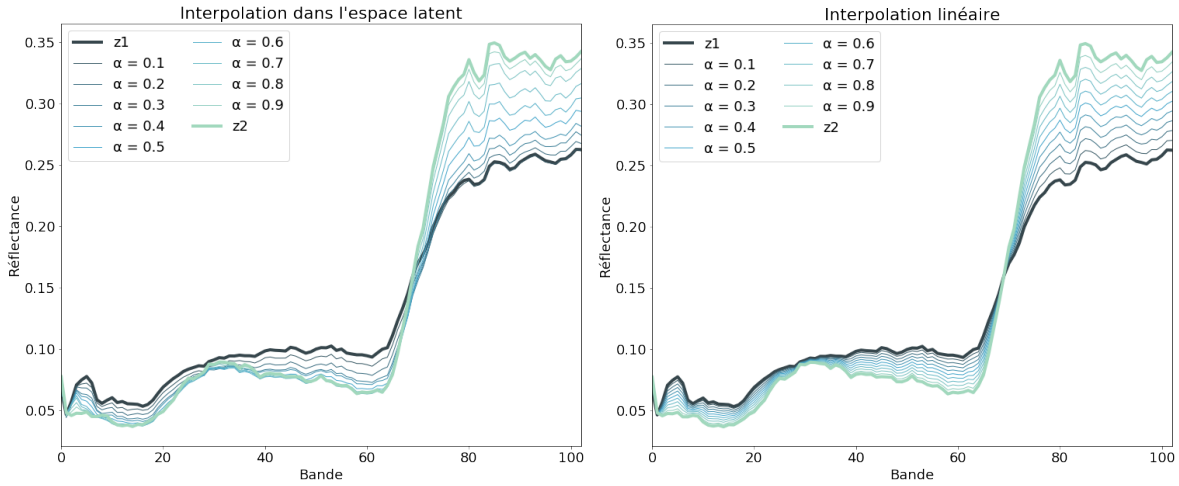
$$\begin{cases} \forall \alpha \in [0,1], \;\; c_\alpha = (1-\alpha)\cdot c_1 + \alpha \cdot c_2 \\ \hat{x}_\alpha = \mathrm{G}(z, c_\alpha) \text{ avec } x_0 = \mathrm{G}(z, c_1) \text{ et } x_1 = \mathrm{G}(z, c_2) \end{cases} \tag{6.4}$$

L'interpolation entre deux points de l'espace latent permet de générer une progression spectrale telle qu'illustrée par la Figure 6.4a. En comparaison à une interpolation linéaire directement effectuée entre les deux signatures spectrales, les échantillons générés ne sont pas régulièrement espacés car le chemin dans l'espace latent encode un réalité un chemin sur la *variété* des spectres. Calculer entre un barycentre entre les deux vecteurs représentant les réponses spectrales n'a pas nécessairement de sens physique, car le vecteur qui en résulte n'est pas nécessairement sur la variété des spectres réels. À l'inverse, les interpolations générées par le GAN approchent fidèlement le chemin sur la variété qui relie les deux extrémités.

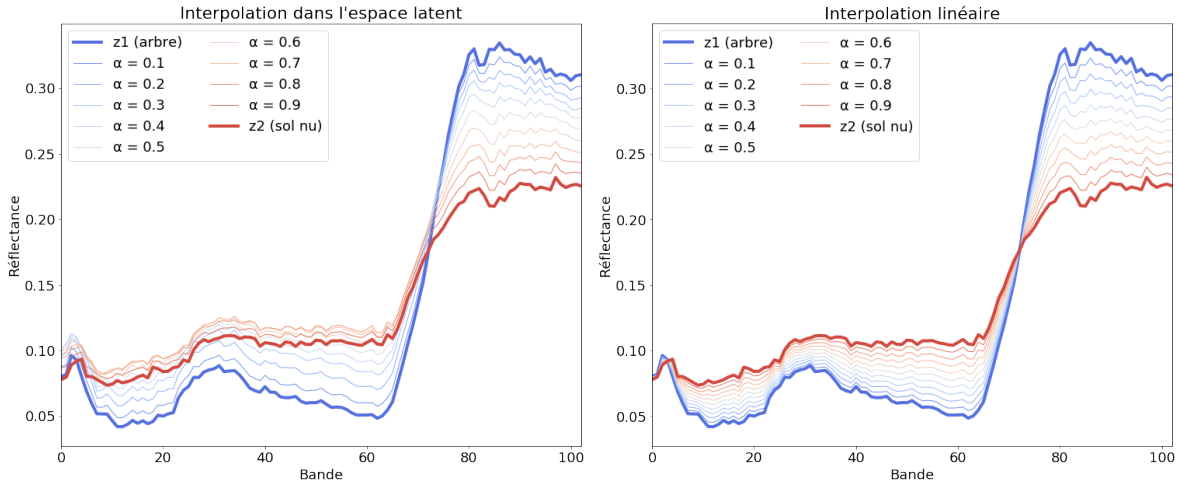
De la même façon, il est possible de générer des mélanges de spectres en interpolant entre les vecteurs de conditionnement, ce qui est illustré par la Figure 6.4b. Les mélanges de matériaux en conditions réelles présentent souvent des propriétés non-linéaires causées par la géométrie du terrain ou des effets de réflexion et d'occlusion. Ici encore, le GAN génère des échantillons dont la présence sur la variété des spectres réels est plausible, tandis que l'interpolation linéaire parcourt un espace arbitraire. Si l'on considère que les mélanges produits par le générateur sont réalistes, alors celui-ci effectue l'inverse de l'opération de démélange. Une approche d'apprentissage par dictionnaire, par inversion de modèle [9] ou par plus proche voisin pourrait alors permettre, à partir d'un spectre réel soupçonné d'être un mélange, de revenir à ses *endmembers* par une cartographie exhaustive de l'espace latent.

6.1.4 Augmentation de données

Les échantillons générés étant plausibles et représentatifs des spectres réels, nous suggérons de les utiliser pour enrichir les jeux de données annotés préexistants. Nous testons cette idée sur plusieurs jeux de données : Indian Pines (aérien, rural), Pavia University (aérien, péri-urbain), Pavia Center (aérien, urbain) et Botswana (satellite, rural). Les résultats en mode supervisé (GAN) et semi-supervisé (ss-GAN) sont détaillés dans le Tableau 6.2. Augmenter le jeu de données à l'aide des faux spectres permet de légèrement augmenter les performances du classifieur.



(a) Interpolation entre deux vecteurs latents de la classe prairie.



(b) Interpolation à vecteur latent constant entre les classes arbre et sol nu.

FIGURE 6.4 – Interpolations dans l'espace latent des spectres Interpoler entre différents vecteurs ou conditionnements de l'espace latent permet d'explorer la variété des spectres de façon continue. Le GAN est entraîné sur le jeu de données Pavia University. α contrôle l'interpolation.

Toutefois, augmenter drastiquement le nombre de faux spectres n'améliore pas plus la classification et finit même par la dégrader. En effet, dans ce cas les échantillons synthétiques deviennent prédominants dans la fonction de coût et dégradent les performances du classifieur, le ramenant vers le cas de la SVM entraînée uniquement sur les faux spectres.

Dans l'absolu, la mise en œuvre de GAN pour la génération de spectres *ex nihilo* et l'augmentation de données n'apporte que des bénéfices légers. Notamment, les GAN ne peuvent qu'approcher la distribution des spectres réellement observés et qu'interpoler à l'intérieur de celle-ci, mais peuvent difficilement générer des nouvelles observations à l'extérieur de la distribution. Puisque la classification consiste à déterminer les frontières entre classes, ce sont donc les échantillons éloignés des centroïdes qui sont les plus informatifs. Ainsi, l'approche semi-supervisée permet de générer des spectres annotés présentant des propriétés statistiques proches des observations non-annotées, et donc d'augmenter la quantité d'information disponible au classifieur, mais l'approche supervisée pure est rapidement limitée. Toutefois, cela a permis de démontrer la capacité des GAN à modéliser des distributions statistiques complexes sans aucune connaissance physique, dans la veine d'autres travaux publiés en parallèle de ces recherches [24]. Cette conclusion devient particulièrement intéressante lorsque l'on considère les efforts passés et actuels investis dans la développement de simulateurs de données hyperspectrales [5]. Ceux-ci utilisent d'une part les mesures en laboratoire des

Tableau 6.2 – Scores d'exactitudes obtenus par un classifieur entièrement connecté à 4 couches sur plusieurs jeux de données en utilisant différentes augmentations de données. Le partage du jeu de données se fait en coupant l'image en deux (s) ou par un échantillonnage aléatoire uniforme de 3% des pixels annotés (r).

Jeu de données	Pavia University		Pavia Center		Botswana		Indian Pines	
Augmentation	3% (r)	50% (s)	3% (r)	50% (s)	3% (r)	50% (s)	3% (r)	50% (s)
∅	92,72	86,22	98,93	96,26	86,90	84,87	79,44	74,00
GAN	92,95	86,47	99,00	96,26	87,72	84,60	80,01	74,81
ss-GAN	93,12	87,20	98,93	96,70	88,40	85,27	80,42	74,58

réflectances de matériaux connus et d'autre part des modèles physiques de capteur et d'atmosphère. Cependant, les modèles introduisent nécessairement des approximations et des simplifications qui ne permettent pas de tenir compte des effets optiques, atmosphériques et électroniques les plus complexes (bruit provoqué par la chauffe des composants, rayons lumineux parasitaires, turbulences…). Une combinaison mêlant approche statistique et physique serait de conditionner les GAN par les spectres produits par ces simulateurs, pour laisser au générateur le soin de modéliser ces phénomènes complexes, afin de rendre les spectres simulés réalistes et de les aligner avec les acquisitions réelles.

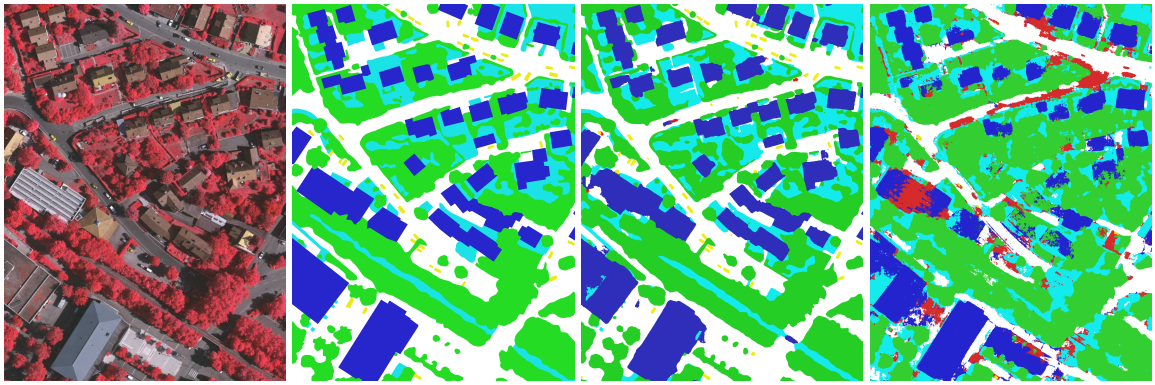
6.2 Cas des données massives

6.2.1 Diversité des scènes

Jusqu'à présent, nous nous sommes intéressés à des jeux de données ne couvrant qu'une seule scène. En effet, les expériences des Chapitres 3 à 5 ont été effectuées sur les villes de Vaihingen et Potsdam, pour une seule ville à la fois. Néanmoins, cela ne correspond pas à un cas applicatif réel. L'observation de la Terre passe par la répétition des acquisitions sur l'ensemble du globe. Par conséquent, il est nécessaire d'évaluer les capacités de généralisation des modèles mis en œuvre dans un cadre géographique varié. Une question naturelle est de savoir comment se comportent les réseaux profonds sur de nouvelles acquisitions. S'il est possible de s'attendre à une dégradation des performances prédictives des modèles, compte-tenu du surapprentissage inhérent à l'entraînement sur une seule scène, il est important d'être en mesure de la quantifier.

Pour ce faire, nous étudions tout d'abord les transferts entre deux scènes similaires : ISPRS Potsdam et ISPRS Vaihingen. Les deux jeux de données sont constitués d'images aériennes EHR recouvrant les canaux IRRV acquises en zone urbaine et ont été annotés pour les mêmes classes d'intérêt. Les villes présentent néanmoins des caractéristiques différentes : Potsdam est six fois plus peuplée que Vaihingen pour une densité de population deux fois plus élevée. Les architectures des bâtiments sont différentes, tout comme l'agencement urbain. En première approche, nous considérons le réseau SegNet entraîné sur les images IRRV de Potsdam au Chapitre 3, que nous appliquons tel quel sur une image de Vaihingen. Les cartes générées sont présentés dans la Figure 6.5. Dans l'ensemble, les grandes composantes de l'image sont retrouvées par le réseau, en particulier les routes et les bâtiments. Néanmoins, on peut constater d'une part une importance confusion des bâtiments avec la classe de rejet (en rouge) mais aussi l'absence de voiture. Cette dernière observation s'explique par la différence de résolution entre les deux jeux de données. Le modèle entraîné sur Potsdam a optimisé ses filtres pour une image à 5 cm/px et est ensuite appliqué sur une image de Vaihingen à 9 cm/px. Le facteur d'échelle ayant changé, les véhicules sont alors plus petits qu'attendu et deviennent invisibles. Dans l'ensemble, l'exactitude de la prédiction sur Vaihingen par un modèle entraîné uniquement sur Potsdam est de 77%, ce qui est nettement inférieur aux résultats que nous avions présenté dans le Chapitre 3. Entraîner un modèle sur une

scène unique semble donc biaiser celui-ci, au détriment de son applicabilité à de nouvelles acquisitions.



Tuile IRRV 21 | Vérité terrain | Prédiction SegNet (Vaihingen) | Prédiction SegNet (Potsdam)

FIGURE 6.5 – Cartes sémantiques prédites sur la tuile 21 de Vaihingen par SegNet entraîné sur Vaihingen et sur Potsdam. Le transfert sans *fine-tuning* depuis Potsdam parvient à identifier les zones principales, mais s'avère significativement moins précis qu'un entraînement direct sur Vaihingen.

Une alternative réaliste consisterait alors à annoter une faible partie des données cible (ici, Vaihingen) et de réaliser un *fine-tuning* du modèle entraîné sur Potsdam. Cela permettrait au modèle d'ajuster ses poids pour tenir compte des nouvelles images sans pour autant entraîner un réseau en entier. Nous considérons donc le même réseau SegNet, qui est spécialisé sur le jeu de données ISPRS Vaihingen de la façon suivante :

— les poids du dernier bloc du décodeur sont optimisés avec un taux d'apprentissage $\alpha = 0,01$,
— le taux d'apprentissage pour les poids restants du décodeur est fixé à $\frac{\alpha}{10}$,
— les poids de l'encodeur sont gelés.

La spécialisation est tentée dans des cas très faiblement annotés, avec seulement un quart de la tuile 3 ou la tuile 3 en entier. Afin de mesurer le gain obtenu par le préentraînement sur Potsdam, nous comparons les performances du modèle spécialisé à celui du SegNet entraîné sur les mêmes images, mais initialisé avec les poids de VGG-16 préappris sur ImageNet. Les résultats sont compilés dans le Tableau 6.3.

TABLEAU 6.3 – Résultats de segmentation sémantique et d'apprentissage par transfert pour le jeu de données ISPRS Vaihingen.

Nombre de tuiles	préentraînement	Routes	Bâtiments	Vég. basse	Arbres	Véhicules	Exac.
1/4	ImageNet	76,6	29,6	0,07	95,1	0,01	54,8
	Potsdam	80,3	55,0	16,0	95,8	43,3	65,5
1	ImageNet	91,8	78,8	50,4	93,5	47,6	81,0
	Potsdam	83,3	91,2	59,4	85,6	60,1	82,8

Lorsque la quantité de tuiles annotées diminue, les performances du modèle baissent significativement. Ce phénomène intervient non seulement lorsque les annotations denses sont moins nombreuses, mais également lorsqu'elles sont rendues parcimonieuses et incompètes. MAGGIOLO et al. [16] ont en effet étudié les performances de FCN entraînés sur Vaihingen en considérant une version grossière de la vérité terrain, diminuée de 60% des pixels annotés. Certains objets sont omis et les autres sont griffonnés sur l'image de façon grossière, reproduisant approximativement leur formes. Le FCN est alors entraîné sur ces pixels et n'apprend pas là où les annotations sont inexistantes ; l'exactitude diminue alors

de près de 20%. Ceci est similaire aux résultats que nous avons obtenus en entraînant un SegNet sur seulement un quart de tuile.

Ainsi, lorsque peu d'annotations sur le domaine cible (ici, Vaihingen) sont disponibles, le préentraînement sur Potsdam permet d'augmenter significativement les performances du réseau lors de l'inférence. En effet, la spécialisation par *fine-tuning* permet de compenser le biais d'apprentissage sur Potsdam et de réduire l'influence des différences environnementales. Dans un cadre applicatif, ces résultats suggèrent qu'une méthode de segmentation pourrait aisément être adaptée après un effort modéré d'annotation sur les nouvelles acquisitions. Toutefois, il est nécessaire de garder à l'esprit que les poids préentraînés sur Potsdam ne sont pas aussi génériques que ceux classiquement utilisés en vision par ordinateur lorsque l'on considère des modèles optimisés sur ImageNet. En effet, ces poids tirent leur expressivité de la grande variété d'images et de classes présentes dans ce jeu de données. Reproduire de telles propriétés en télédétection nécessite alors de constituer un jeu de données à grande échelle avec une variabilité importante, tant en sémantique qu'en type de scènes observées.

6.2.2 MiniFrance



Répartition des villes | Orthoimage (Nice) | Annotations (UrbanAtlas)

Figure 6.6 – Présentation du jeu de données MiniFrance.

La création d'un équivalent à ImageNet pour la télédétection nécessite de rassembler et d'annoter une grande quantité de données. Toutefois, si l'étiquetage d'images pour la classification est une tâche rapide, les annotations denses pour la segmentation sont elles nettement plus longues à réaliser. En particulier, dans le cas des images de télédétection, distinguer certains types d'objet nécessite parfois une habitude et une expertise de photointerprétation hors de portée des stratégies d'annotations reposant sur le travail de nombreux collaborateurs ou volontaires [4].

Nous choisissons donc d'utiliser des sources de données semi-manuelles combinant classification automatique et correction par un photointerprète, dont la précision est vraisemblablement inférieure à celle de l'humain mais couvrant une surface bien plus importante. L'apprentissage sur des annotations imparfaites et incomplètes peut nécessiter de mettre en œuvre des techniques spécifiques [14], toutefois nous considérons une première approche directe simplement supervisée. Dans un premier temps, nous choisissons de restreindre l'étendue du jeu de données à l'échelle de la France. En effet, ce pays présente un climat tempéré, avec une grande variété d'environnements (montagnes, littoraux, forêts, cultures, agglomérations urbaines…) et plusieurs sources d'annotations exploitables.

Nous collectons un jeu de données à large échelle sur la France métropolitaine. Nous utilisons la BD ORTHO de l'IGN comme source d'images aériennes à une résolution de 50 cm/px. En particulier, afin de permettre la réutilisation du jeu de données, nous considérons les acquisitions réalisées entre 2012 et 2015 placées sous Licence ouverte 2.0, ce qui correspond à 25 départements. Les données sont fournies sous la forme de tuiles RVB de dimensions 10 000 px × 10 000 px, c'est-à-dire de 25 km^2. Les images sont initialement

fournies sous format JPEG2000 et nous les convertissons en GeoTIFF encodés en entiers sur 8 bits.

En parallèle, nous récupérons les annotations du projet Copernicus *Urban Atlas* 2012 [2]. Il s'agit d'une carte européenne d'occupation des sols pour 17 classes urbaines et 10 classes rurales couvrant les agglomérations de l'Union Européenne de plus de 30 000 habitants. Les annotations sont réalisées de façon semi-automatique par deux photo-interprètes à partir d'images satellitaires acquises durant l'année 2012, principalement SPOT-5. En France, cela concerne 82 aires urbaines et leur périphérie. L'intersection de ces données avec les images de la BD ORTHO permet d'identifier 16 agglomérations et leurs alentours, pour des images de 2012 à 2014 afin de limiter les écarts avec la référence *Urban Atlas*. Les villes identifiées sont listées dans le Tableau 6.4. Les *shapefiles* sont ainsi rasterisés pour chaque image de la BD ORTHO considérée afin de générer des vérités terrain de segmentation sémantique. Profitant de la structure hiérarchique de la taxonomie *Urban Atlas*, nous regroupons les différentes étiquettes en 14 catégories d'occupation des sols détaillées dans le Tableau 6.5.

En annotations annexes, exploitables pour le futur, nous considérons également le cadastre français des bâtiments, également sous licence ouverte. En particulier, nous intégrons et rasterisons tuile par tuile le cadastre au format *shapefile* produit par la mission Etalab [3]. Nous excluons des annotations les bâtiments ajoutés après le 1er janvier 2015, qui ne sont normalement pas encore présents pas dans les images de la BD ORTHO que nous considérons.

Le jeu de données complet, que nous intitulons *MiniFrance*, est illustré par la Figure 6.6. Les 16 agglomérations se trouvent majoritairement dans l'ouest de la France, mais le sud-est, le centre et le nord sont également représentés. 8 villes sont utilisées pour l'apprentissage et les 8 restantes sont conservées pour l'évaluation.

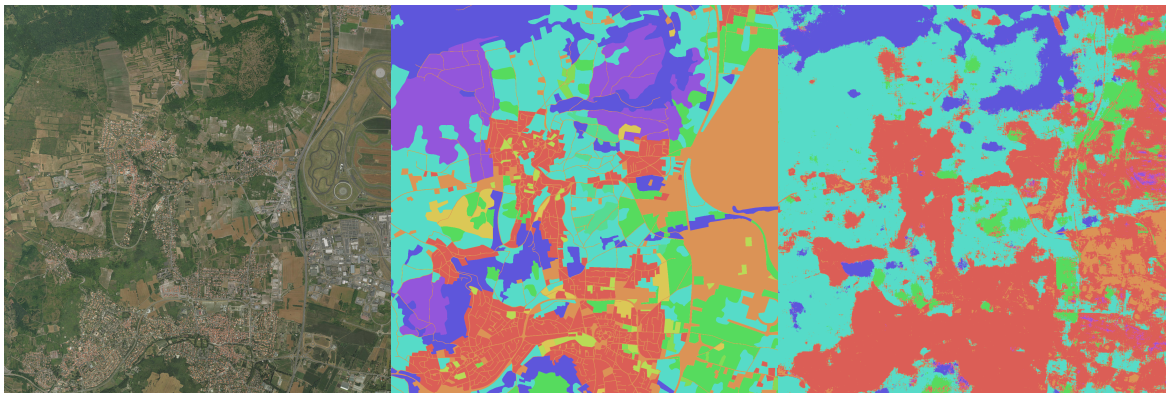
Tableau 6.4 – Liste des villes présentes dans MiniFrance.

	Conurbation	Tuiles	% pixels	Couleur
Entraînement	Nice	170	8,01%	
	Nantes, Saint-Nazaire	226	10,65%	
	Le Mans	107	5,04%	
	Lorient	68	3,20%	
	Brest	88	4,14%	
	Caen	126	5,94%	
	Dunkerque, Calais, Boulogne-sur-Mer	150	7,07%	
	Saint-Brieuc	71	3,34%	
Évaluation	Marseille, Martigues	162	7,63%	
	Rennes	196	9,24%	
	Angers	123	5,79%	
	Quimper	79	3,72%	
	Vannes	73	3,44%	
	Clermont-Ferrand	150	7,07%	
	Lille, Arras, Lens, Douai, Hénin-Beaumont	275	12,96%	
	Cherbourg	57	2,68%	

Afin d'établir un premier résultat de référence sur MiniFrance, nous entraînons un modèle SegNet pour la segmentation sémantique des 14 classes d'occupation des sols de *Urban Atlas* [4]. Les hyperparamètres sont identiques à ceux présentés dans le Chapitre 3. Notons que les classes considérées sont significativement plus complexes à identifier que celles des jeux

2. Urban Atlas : https://land.copernicus.eu/local/urban-atlas
3. Cadastre Etalab : https://cadastre.data.gouv.fr/datasets/cadastre-etalab
4. En pratique, seulement 12 sont à prendre en compte car les classes "Forêts" et "Vergers" ne sont pas représentées dans MiniFrance.

de données manipulés jusqu'ici. En effet, il s'agit d'usages des sols, un concept abstrait qui n'est pas lié à un objet spécifique mais à un quartier. Distinguer les bâtiments commerciaux des bâtiments résidentiels nécessite de fait une séparation plus fine entre les types d'objets. En outre, les annotations de *Urban Atlas* sont moins précises et contiennent potentiellement plus d'erreurs et de différences avec le contenu des images que celles réalisées pour les jeux de données ISPRS. Enfin, la diversité des villes et des environnements représentés dans MiniFrance nécessitent que le modèle soit robuste aux variations d'apparence et puisse généraliser d'une agglomération à une autre. Du point de vue purement statistique, MiniFrance exhibe en effet une variance nettement plus élevée que Vaihingen, avec d'importantes variations entre les villes qui le composent, comme compilé dans les Tableaux 6.7a et 6.7b.



(a) Image RVB (b) Vérité terrain (c) Prédiction

Figure 6.7 – Exemple de carte sémantique obtenue sur MiniFrance. Ici, un extrait en banlieue de Clermont-Ferrand.

Les métriques de segmentation sémantique obtenues par un SegNet entraîné sur MiniFrance sont compilées dans le Tableau 6.6. Comme attendu, les scores sont relativement faibles, le score F_1 variant significativement entre les différentes classes. Les usages des sols résidentiels, commerciaux et agricoles sont relativement bien identifiés tandis que les classes les plus rares (mines, installations sportives, plans d'eau...) ne sont pas apprises. Dans l'ensemble, cette première expérience semble indiquer que si la discrimination grossière entre les différents types d'occupation des sols (bâti, végétation, cultures) est abordable, la classification de l'usage des sols est elle plus complexe. Un exemple de carte produite par le modèle sur une tuile de MiniFrance est donné dans la Figure 6.7. Ces résultats demeurent donc encourageants compte-tenu de la difficulté de la tâche. Notamment, la variété des agglomérations observées et l'échelle jusqu'ici inégalée du jeu de données en font un défi particulièrement intéressant à relever pour l'interprétation d'images de télédétection dans le futur.

Une partie des travaux présentés dans ce chapitre ont été le sujet d'une publication en conférence :

— Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « Generative Adversarial Networks for Realistic Synthesis of Hyperspectral Samples ». Dans : *2018 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. Juil. 2018, p. 5091-5094

Tableau 6.5 – Taxonomie des types d'occupation des sols de UrbanAtlas 2012.

Code	Occupation du sol	Classe	Couleur
10000	Surfaces artificielles	1	
11000	Tissu urbain	1	
11100	Tissu urbain dense	1	
11200	Tissu urbain discontinu	1	
11210	Tissu urbain discontinu dense	1	
11220	Tissu urbain discontinu moyennement dense	1	
11230	Tissu urbain discontinu peu dense	1	
11240	Tissu urbain discontinu très peu dense	1	
11300	Structures isolées	1	
12100	Bâtiments industriels, commerciaux, publics, militaires, privés ou de transports	2	
12200	Réseaux routiers et ferrés et terrains associés	2	
12210	Réseau autoroutier et terrains associés	2	
12220	Autres routes et terrains associés	2	
12230	Voies ferrées et terrains associés	2	
12300	Zones portuaires	2	
12400	Aéroports	2	
13000	Mines, sites de construction et d'enfouissement	3	
13100	Mines et zones d'enfouissement	3	
13300	Sites de construction	3	
13400	Terrain non-occupé	3	
14000	Aires artificiellement végétalisées non-agricoles	4	
14100	Espaces verts urbains	4	
14200	Installations sportives et de loisirs	4	
20000	Aires agricoles, aires semi-naturelles et zones humides	5	
21000	Terres arables (cultures annuelles)	5	
22000	Cultures permanentes	6	
23000	Pâturages	7	
24000	Modèles de culture complexes et hétérogènes	8	
25000	Vergers	9	
31000	Forêts	10	
32000	Associations végétales herbacées	11	
33000	Espaces ouverts avec peu ou pas de végétation	12	
40000	Zones humides	13	
50000	Plans d'eau	14	

Tableau 6.6 – Performances de segmentation sémantique d'un modèle SegNet sur MiniFrance (scores F_1 par classe et exactitude globale).

Classe	1	2	3	4	5	6	7	8	11	12	13	14	Global
Exactitude	50.89	41.39	0.00	0.00	0.00	56.74	0.84	52.26	64.65	8.70	1.23	0.01	51.97

Tableau 6.7 – Comparaison des statistiques au niveau pixel entre Vaihingen et MiniFrance.

(a) Statistiques au niveau pixel par canal pour Vaihingen.

Ensemble	Moyenne ± écart-type		
	Infrarouge	Rouge	Vert
Entraînement	120,30± 54,47	81,64± 38,58	80,52± 36,62
Évaluation	118,07± 56,23	80,69± 41,75	79,70± 40,37
Total	119,74± 54,93	81,40± 39,40	80,32± 37,59

(b) Statistiques au niveau pixel par canal pour MiniFrance.

Agglomération	Moyenne ± écart-type		
	Rouge	Vert	Bleu
Nice	87,44± 67,04	95,70± 60,50	76,11± 60,19
Nantes, Saint-Nazaire	126,05± 46,64	132,81± 35,85	109,25± 38,09
Le Mans	108,06± 57,10	122,98± 44,05	85,93± 39,32
Lorient	89,43± 62,12	100,80± 53,21	87,12± 52,82
Brest	120,53± 76,08	134,98± 62,98	107,72± 62,76
Caen	127,55± 56,26	134,88± 40,76	114,36± 41,51
Dunkerque, Calais, Boulogne-sur-Mer	133,43± 66,10	138,65± 55,43	123,01± 56,93
Saint-Brieuc	116,91± 61,63	128,37± 50,72	105,12± 52,52
Marseille, Martigues	102,43± 62,58	109,71± 55,51	95,53± 57,45
Rennes	94,82± 46,42	110,57± 36,62	87,34± 28,17
Angers	123,04± 48,27	124,21± 33,14	97,28± 34,77
Quimper	115,04± 72,31	127,73± 58,71	104,76± 56,80
Vannes	75,70± 43,08	84,33± 32,72	68,00± 27,94
Clermont-Ferrand	93,74± 33,58	101,79± 25,79	77,41± 20,21
Lille, Arras, Lens, Douai, Hénin-Beaumont	120,14± 58,20	121,78± 47,45	100,94± 48,30
Cherbourg	123,90± 62,51	127,77± 57,14	114,54± 60,34
Entraînement	115,30± 62,90	124,33± 52,42	101,94± 52,78
Évaluation	106,81± 55,59	113,91± 45,41	93,00± 44,49
Total	110,83± 59,32	118,85± 49,14	97,24± 48,80

Références

[1] Jacopo Acquarelli et al. « Convolutional Neural Networks and Data Augmentation for Spectral-Spatial Classification of Hyperspectral Images ». Dans : (15 nov. 2017). arXiv : 1711.05512 [cs]. url : http://arxiv.org/abs/1711.05512 (cf. p. 142).

[2] Martin Arjovsky, Soumith Chintala et Léon Bottou. « Wasserstein Generative Adversarial Networks ». Dans : *Proceedings of the International Conference on Machine Learning (ICML)*. International Conference on Machine Learning. 17 juil. 2017, p. 214-223. url : http://proceedings.mlr.press/v70/arjovsky17a.html (cf. p. 143).

[3] Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « Generative Adversarial Networks for Realistic Synthesis of Hyperspectral Samples ». Dans : *2018 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. Juil. 2018, p. 5091-5094 (cf. p. 153).

[4] Adela Barriuso et Antonio Torralba. « Notes on Image Annotation ». Dans : (12 oct. 2012). arXiv : 1210.3448 [cs]. url : http://arxiv.org/abs/1210.3448 (cf. p. 151).

[5] Anko Börner et al. « SENSOR : A Tool for the Simulation of Hyperspectral Remote Sensing Systems ». Dans : *ISPRS Journal of Photogrammetry and Remote Sensing* 55.5-6 (1er mar. 2001), p. 299-312. issn : 0924-2716. doi : 10.1016/S0924-2716(01)00022-3. url : https://www.sciencedirect.com/science/article/pii/S0924271601000223 (cf. p. 148).

[6] Yushi Chen et al. « Deep Feature Extraction and Classification of Hyperspectral Images Based on Convolutional Neural Networks ». Dans : *IEEE Transactions on Geoscience and Remote Sensing* 54.10 (oct. 2016), p. 6232-6251. issn : 0196-2892. doi : 10.1109/TGRS.2016.2584107 (cf. p. 142).

[7] Amir Abbas Davari et al. « GMM-Based Synthetic Samples for Classification of Hyperspectral Images With Limited Training Data ». Dans : *IEEE Geoscience and Remote Sensing Letters* 15.6 (juin 2018), p. 942-946. issn : 1545-598X. doi : 10.1109/LGRS.2018.2817361 (cf. p. 142).

[8] David A. van Dyk et Xiao-Li Meng. « The Art of Data Augmentation ». Dans : *Journal of Computational and Graphical Statistics* (1er jan. 2012). doi : 10.1198/10618600152418584. url : http://amstat.tandfonline.com/doi/abs/10.1198/10618600152418584 (cf. p. 142).

[9] Ian Gemp et al. « Inverting Variational Autoencoders for Improved Generative Accuracy ». Dans : *NIPS Workshop on Advances in Approximate Bayesian Inference*. 2017. arXiv : 1608.05983. url : http://arxiv.org/abs/1608.05983 (cf. p. 142, 147).

[10] Ian Goodfellow et al. « Generative Adversarial Nets ». Dans : *Proceedings of the Neural Information Processing Systems (NIPS)*. NIPS. 2014, p. 2672-2680. url : http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf (cf. p. 142, 143).

[11] Ishaan Gulrajani et al. « Improved Training of Wasserstein GANs ». Dans : *Proceedings of the Neural Information Processing Systems (NIPS)*. NIPS. 2017, p. 5769-5779. url : http://papers.nips.cc/paper/7159-improved-training-of-wasserstein-gans.pdf (cf. p. 143).

[12] Alex Krizhevsky, Ilya Sutskever et Geoffrey E. Hinton. « ImageNet Classification with Deep Convolutional Neural Networks ». Dans : *Proceedings of the Neural Information Processing Systems (NIPS)*. NIPS. 2012, p. 1097-1105. url : http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf (cf. p. 142).

[13] Hyungtae Lee et Heesung Kwoon. « Contextual Deep CNN Based Hyperspectral Classification ». Dans : *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. IGARSS. Beijing, juil. 2016, p. 3322-3325. DOI : 10.1109/IGARSS.2016.7729859 (cf. p. 142).

[14] Zhiwu Lu et al. « Learning from Weak and Noisy Labels for Semantic Segmentation ». Dans : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.3 (mar. 2017), p. 486-500. ISSN : 0162-8828. DOI : 10.1109/TPAMI.2016.2552172 (cf. p. 151).

[15] Andrew L. Maas, Awni Y. Hannun et Andrew Y. Ng. « Rectifier Nonlinearities Improve Neural Network Acoustic Models ». Dans : *ICML Workshop on Deep Learning for Audio, Speech and Language Processing*. 2013 (cf. p. 144).

[16] Luca Maggiolo et al. « Improving Maps from CNNs Trained with Sparse, Scribbled Ground Truths Using Fully Connected CRFs ». Dans : *2018 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. 2018 IEEE International Geoscience and Remote Sensing Symposium (IGARSS). Juil. 2018, p. 2103-2106 (cf. p. 150).

[17] Konstantinos Makantasis et al. « Deep Supervised Learning for Hyperspectral Data Classification through Convolutional Neural Networks ». Dans : *Geoscience and Remote Sensing Symposium (IGARSS), 2015 IEEE International*. Geoscience and Remote Sensing Symposium (IGARSS), 2015 IEEE International. Juil. 2015, p. 4959-4962. DOI : 10.1109/IGARSS.2015.7326945 (cf. p. 142).

[18] Augustus Odena, Christopher Olah et Jonathon Shlens. « Conditional Image Synthesis with Auxiliary Classifier GANs ». Dans : *International Conference on Machine Learning*. International Conference on Machine Learning. 17 juil. 2017, p. 2642-2651. URL : http://proceedings.mlr.press/v70/odena17a.html (cf. p. 144).

[19] Fabian Pedregosa et al. « Scikit-Learn : Machine Learning in Python ». Dans : *Journal of Machine Learning Research* 12 (Oct 2011), p. 2825-2830. ISSN : ISSN 1533-7928. URL : http://www.jmlr.org/papers/v12/pedregosa11a.html (cf. p. 145).

[20] Tim Salimans et al. « Improved Techniques for Training GANs ». Dans : *Proceedings of the Neural Information Processing Systems (NIPS)*. NIPS. 2016, p. 2234-2242. URL : http://papers.nips.cc/paper/6125-improved-techniques-for-training-gans.pdf (cf. p. 145).

[21] Viktor Slavkovikj et al. « Hyperspectral Image Classification with Convolutional Neural Networks ». Dans : *Proceedings of the 23rd ACM International Conference on Multimedia*. ACM Press, 2015, p. 1159-1162. ISBN : 978-1-4503-3459-4. DOI : 10.1145/2733373.2806306. URL : http://dl.acm.org/citation.cfm?doid=2733373.2806306 (cf. p. 142).

[22] Tijmen Tielman et Geoffrey Hinton. *Lecture 6.5—RmsProp : Divide the Gradient by a Running Average of Its Recent Magnitude*. 2012 (cf. p. 145).

[23] Lloyd Windrim et al. « Hyperspectral CNN Classification with Limited Training Samples ». Dans : (28 nov. 2016). arXiv : 1611.09007 [cs]. URL : http://arxiv.org/abs/1611.09007 (cf. p. 142).

[24] Lin Zhu et al. « Generative Adversarial Networks for Hyperspectral Image Classification ». Dans : *IEEE Transactions on Geoscience and Remote Sensing* 56.9 (sept. 2018), p. 5046-5063. ISSN : 0196-2892. DOI : 10.1109/TGRS.2018.2805286 (cf. p. 148).

Chapitre 7

Spatialisation des prédictions pixelliques

Homo sapiens is about pattern recognition, he says. Both a gift and a trap.

— William Gibson (Pattern Recogntion, 2002)

Sommaire

7.1 *Segment-before-detect* 160
7.1.1 Régions et objets 160
7.1.2 Segmentation de véhicules 162
7.1.3 Détection de véhicules 163
7.1.4 Classification de véhicules 167
7.2 Segmentation sémantique par régression des cartes de distances 171
7.2.1 Annotations sémantiques et cartes de distances 171
7.2.2 Apprentissage multitâche 173
7.2.3 Validation expérimentale 174

Résumé du chapitre :

Jusqu'à présent, nous nous sommes intéressés à la segmentation sémantique sous l'angle de la classification pixellique dense. Bien que les modèles aient pris en compte le contexte spatial, la fonction de coût ne s'intéressait qu'à réduire l'erreur moyenne sur les pixels. Toutefois, la compréhension de scènes nécessite bien d'extraire et de manipuler des concepts liés aux objets et aux relations entre eux, et non des pixels.

Ce chapitre s'intéresse ainsi aux extensions des modèles de réseaux de neurones permettant d'exploiter la structure spatiale des objets d'intérêt dans les images.

Dans un premier temps, nous cherchons à structurer *a posteriori* les prédictions pixelliques issues des FCN que nous avons entraînés. En particulier, nous montrons qu'il est possible d'aisément structurer les prédictions pixelliques générées par les FCN à des fins de détection et d'identification du type des véhicules dans des images aériennes.

Dans un second temps, nous étudions les représentations alternatives de la vérité terrain permettant d'intégrer des notions spatiales dans l'optimisation des réseaux de neurones. Plus précisément, nous introduisons une fonction de coût alternative opérant sur une version continue des annotations sémantiques obtenue par transformée de distances. Cette approche en régression des cartes de distances est complémentaire au modèle de classification. En pratique, nous proposons d'entraîner un réseau multitâche réalisant simultanément la régression des cartes de distances euclidiennes et la classification des pixels. Cette méthode permet de coupler géométrie et sémantique dans la fonction de coût et régularise ainsi les segmentations.

7.1 Segment-before-detect

7.1.1 Régions et objets

La détection et la reconnaissance de véhicules sont deux problèmes classiques en télédétection. Non seulement la localisation et l'identification des véhicules interviennent dans le cadre de la surveillance et de l'interprétation de scènes, mais ces informations permettent également d'isoler les parties non mobiles d'une image pour mieux en comprendre la géométrie [31]. De nombreux travaux ont étudié la détection automatique de véhicules dans des images THR avec une large variété de méthodes, allant des descripteurs HOG couplés à des SVM [37, 16, 27] aux modèles 3D d'estimation de pose [24] en passant par les modèles à parties déformables [43] ou les mélanges de modèles invariants par rotation [44]. L'apprentissage profond et notamment les CNN ont également été appliqués à cette tâche [10, 39]. Récemment, la majorité des méthodes utilisent des réseaux spécifiquement conçus pour la détection d'objet, comme Faster-RCNN [47] dans les travaux de Sommer, Schuchert et Beyerer [52] ou YOLO [46] dans ceux de Van Etten [59]. Cependant, peu de travaux s'intéressent à réaliser simultanément détection et reconnaissance, en dépit de l'introduction de la base de données *Vehicle Detection in Aerial Imagery* (VEDAI) par Razakarivony et Jurie [45] et d'une première approche utilisant des caractéristiques expertes pour classifier divers véhicules à partir d'images IRRVB. Certains travaux plus anciens s'étaient intéressés à cette problématique sous la forme d'une segmentation multiéchelle et de l'utilisation de règles de logique floue [22], puis d'analyse discriminante linéaire [14], ou encore de classifieurs experts appliqués à une segmentation par détection de contours [55]. En particulier, Eikvil, Aurdal et Koren [14] montrent que segmenter l'image avant de réaliser la détection des véhicules permet d'éliminer de nombreux faux positifs.

Dans cette perspective, nous proposons donc d'étudier comment, à partir des approches de segmentation sémantique par apprentissage profond étudiées au Chapitre 3, aboutir à des méthodes de détection et de classification de véhicules par raffinements successifs. Nous présentons dans cette partie une méthode complète de segmentation pour la détection intitulée *segment-before-detect*. Au-delà des boîtes englobantes habituellement utilisées en détection, nous allons montrer comment prédire la forme et le type des instances individuelles des véhicules présents dans une image aérienne.

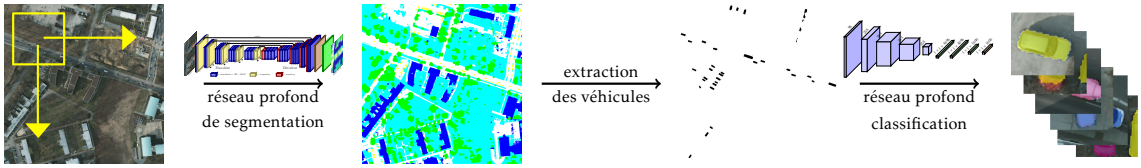
Notre méthode *segment-before-detect* permet d'extraire et de classifier des véhicules à partir d'images THR et consiste en trois étapes, illustrées dans la Figure 7.1 :

1. Segmentation sémantique et inférence du masque de véhicules au niveau pixel par un FCN,
2. Détection d'instance par régression des enveloppes convexes des composantes connexes,
3. Classification des objets identifiés à l'aide d'un CNN.

Détection de petits objets

Comme nous l'avons vu dans le Chapitre 3, un réseau profond de type SegNet est suffisamment précis pour prédire des véhicules individuels dans des images haute résolution (< 50 cm/px). Dans ce cas, il suffit alors de réaliser une extraction des composantes connexes pour obtenir les instances individuelles des véhicules présents dans l'image. Pour chaque composante connexe, le calcul de la boîte englobante l'entourant est alors immédiat.

Cependant, les prédictions issues de SegNet sont susceptibles d'être bruitées. En particulier, les CNN appliqués à l'observation de la Terre tendent à produire des transitions inter-classes imprécises [35]. Par conséquent, nous éliminons une grande partie des faux positifs et séparons les véhicules susceptibles d'avoir été prédits comme appartenant à la même composante connexe en appliquant une ouverture morphologique au masque sé-



Fenêtre glissante sur l'image RVB | Carte sémantique | Masque des véhicules | Véhicules classifiés

FIGURE 7.1 – Illustration de la méthode *segment-before-detect* pour la segmentation, détection et classification de véhicules.



Image RVB et segmentation | Masque des véhicules | Masque après ouverture | Composantes connexes

FIGURE 7.2 – Processus de localisation des instances de véhicules par ouverture morphologique et extraction des composantes connexes.

mantique obtenu par SegNet (cf. Figure 7.2) [1]. Nous éliminons ensuite les objets dont la surface est inférieure à un seuil strict pour supprimer les fausses alarmes dues aux erreurs de segmentation, comme les bouches de ventilation sur les toits ou certains éléments du mobilier urbain. Bien que simple, nous allons voir que cette approche permet d'augmenter significativement les performances en détection de SegNet.

Reconnaissance de véhicules par CNN

En admettant que nous sommes en mesure de localiser les véhicules, nous cherchons dès lors à en identifier le type. Ainsi, nous cherchons à déterminer si le véhicule est une voiture, un camion, une camionette, etc. Il s'agit d'un problème classique de reconnaissance d'objet que les CNN peuvent aisément résoudre. Nous appliquons donc l'approche classique consistant à spécialiser [38, 64] un modèle de CNN préentraîné sur la base de données ImageNet [49] pour la reconnaissance de véhicules.

Nous comparons en particulier les modèles les plus utilisés dans la littérature pour la classification de petites images ($\simeq$30 px $\times$ 30 px) : LeNet [32], AlexNet [29] et VGG-16 [51].

Notre objectif est d'entraîner ce classifieur sur une grande base de données de véhicules et de l'appliquer sur des données issues d'une scène spécifique. Nous risquons donc d'être confrontés au problème de surapprentissage. En effet, nous cherchons à transférer des connaissances d'un jeu de données à un autre. Pour améliorer la généralisabilité du modèle, deux techniques peuvent être employées : l'adaptation de domaine et l'augmentation de données. L'adaptation de domaine cherche à minimiser les différence entre le jeu de données d'apprentissage et celui d'inférence. L'augmentation de données cherche quant à elle à générer de nouvelles images d'entraînement synthétiques pour améliorer la robustesse du classifieur.

Nous proposons ainsi de normaliser l'ensemble des véhicules pour que ceux-ci présentent le même azimuth, c'est-à-dire que toutes les images présentent un véhicule dont la direction

1. La perte des contours exacts n'est pas critique dans la mesure où la classification se fera sur un *patch* centré sur la composante connexe.



identité rotation 90° rotation 180° sym. horizon. sym. verticale zoom translation

Figure 7.3 – Augmentation de données sur un véhicule de la base VEDAI.

principale est horizontale. Durant l'apprentissage, nous utilisons les boîtes englobantes pour estimer la direction principale du véhicule puis nous appliquons une rotation pour l'alignement. Durant l'inférence, nous utilisons la composante connexe du masque correspondant au véhicule pour réaliser cette opération.

Enfin, nous augmentons le jeu de données d'apprentissage en appliquant diverses opérations géométriques pour augmenter la variété des échantillons : translations (±10 px), zooms (jusqu'à 1,25×), rotations (90°, 180°et 270°) et symétries axiales, comme illustrées par la Figure 7.3. Lorsque la stratégie de réalignement est utilisée, seule la rotation à 180° est appliquée.

7.1.2 Segmentation de véhicules

Afin d'entraîner un CNN pour la classification de véhicules, nous devons utiliser une base d'apprentissage suffisamment grande. Pour ce faire, nous utilisons le jeu de données VEDAI [45] (cf. Annexe A.1.5), contenant de nombreuses annotations de véhicules dans des images aériennes. VEDAI est utilisé pour entraîner le CNN de reconnaissance de véhicules, qui sera appliqué en inférence sur le jeu de données ISPRS Potsdam, noté Potsdam par la suite (cf. Annexe A.1.1). Les résultats sur VEDAI sont obtenus par validation croisée en utilisant 2/3 des images pour l'entraînement et 1/3 pour l'inférence.

Afin de valider notre méthode, nous utilisons dans un premier temps le jeu de données ISPRS Potsdam (cf. Figure 7.4) sur lequel nous avons manuellement annoté les véhicules dans quatre sous-catégories : voiturfes, vans, camions et pick-ups. Les véhicules ayant été initialement annotés dans la classe de rejet (engins de chantier, notamment) dans la vérité terrain originale ne sont pas pris en compte. Comme indiqué dans le Tableau 7.1, le jeu de données est largement dominé par les voitures (94% des véhicules).

Nous entraînons un SegNet pour la segmentation sémantique sur ce jeu de données et nous appliquons le CNN entraîné sur VEDAI pour la reconnaissance de véhicules. Les résultats sont obtenus par validation croisée utilisant 18 tuiles pour l'entraînement et 6 tuiles pour la validation. La résolution spatiale est interpolée à 12,5 cm/px pour correspondre avec celle de VEDAI au lieu des 5 cm/px initiaux.

Dans un second temps, nous utilisons le jeu de données NZAM/ONERA Christchurch, noté Christchurch par la suite (cf. Annexe A.1.6). Pour rappel, la vérité terrain de ce jeu de données est moins précise que celle de Potsdam et se rapproche plus des annotations par polygones englobants habituellement disponibles en détection, comme illustré par la Figure 7.4. Nous étendons également cette vérité terrain en déclinant les véhicules en quatre types : voitures, camions, vans et pick-ups. À nouveau, le jeu de données est dominé par la classe voiture (cf. Tableau 7.1).

Le jeu de données NZAM/ONERA Christchurch contenant seulement des polygones englobants, susceptibles de s'intersecter, pour les arbres, les bâtiments et les véhicules, nous transformons ces annotations en cartes sémantiques. Pour ce faire, nous définissons quatre classes d'intérêt : arrière-plan, bâtiments, végétation et véhicules. Nous construisons la vérité terrain dense en étiquetant d'abord les pixels appartenant à une boîte englobante de bâtiment, puis ceux appartenant à des véhicules et finalement à la végétation. Les pixels restants sont étiquetés comme arrière-plan. Cet ordre permet de prendre en compte la présence de certains véhicules sur des parkings aériens et la présence de végétation arborescente pouvant masquer certaines voitures. Pour prendre en compte l'incertitude sur les boîtes englobantes, les pixels



(a) ISPRS Potsdam (b) NZAM/ONERA Christchurch

Figure 7.4 – Exemples d'annotations dans les jeux de données considérés.

Tableau 7.1 – Nombre de véhicules par classe dans les différents jeux de données.

Jeu de données	Voiture	Camion	Van	Pick-up	Bateau	Camping-car	Autres	Avion	Tracteur
VEDAI	1340	300	100	950	170	390	200	47	190
ISPRS Potsdam	1990	33	181	40	-	-	-	-	-
Christchurch	2267	73	120	90	-	-	-	-	-

appartenant à un disque de rayon 5 px autour des frontières sont ignorés, soit environ 60 cm.

Nous entraînons un SegNet pour la segmentation sémantique sur ce jeu de données et nous appliquons le CNN entraîné sur VEDAI pour la reconnaissance de véhicules. Les résultats sont obtenus par validation croisée utilisant 3 tuiles pour l'entraînement et 1 tuile pour la validation. La résolution spatiale est interpolée à 12,5 cm/px pour correspondre avec celle de VEDAI au lieu des 10 cm/px initiaux. Les hyperparamètres habituels décrits au Chapitre 3 sont également repris ici.

Segmentation sémantique

Nous détaillons dans le Tableau 7.2 les scores F_1 et l'exactitude globale du SegNet entraîné sur Potsdam. Il est important de noter que les résultats obtenus à résolution de 12,5 cm/px sont similaires à ceux obtenues à 5 cm/px. Un exemple qualitatif de segmentation est illustré dans la Figure 7.5.

Comme illustré dans la Tableau 7.3, un SegNet entraîné sur NZAM/ONERA Christchurch atteint un score F_1 de 61,9% sur les véhicules, ce qui est acceptable compte-tenu de l'imprécision des annotations en comparaison de celles de Potsdam. Ce constat est intéressant dans la mesure où il prouve qu'il est possible d'apprendre des modèles de segmentation sémantique, même en présence de simples boîtes englobantes originellement prévues pour entraîner des détecteurs. L'inférence sur une tuile de Christchurch nécessite environ 120 secondes sur un GPU NVIDIA Tesla K20c. Un exemple de segmentation est illustré dans la Figure 7.5.

7.1.3 Détection de véhicules

Nous appliquons sur les deux jeux de données une ouverture morphologique d'un rayon de 3 px (≃ 35 cm d'incertitude sur les formes de véhicules prédites) pour isoler des véhicules ayant pu être prédits dans la même composante connexe. Nous supprimons également les composantes connexes couvrant moins de 1,5 m^2 (100 px). Une voiture citadine couvre environ 4 m^2, et nous considérons que les occlusions peuvent recouvrir jusqu'à 60% du véhicule. Nous extrayons ensuite les composantes connexes dans le masque des véhicules et nous calculons la boîte englobante de chaque composante. Pour le jeu de données ISPRS

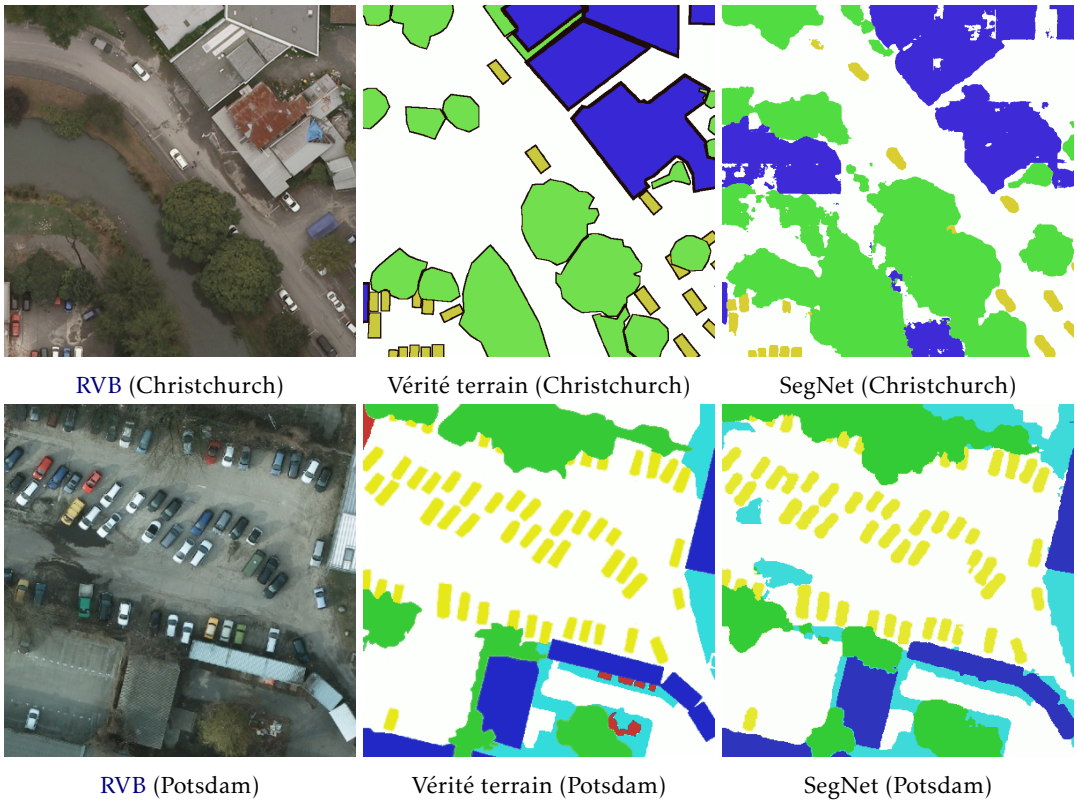


FIGURE 7.5 – Exemples de segmentations obtenues sur Potsdam et Christchurch.
Légende : blanc : routes, bleu : bâtiments, cyan : végétation basse, vert : arbres, jaune : véhicules, rouge : autre.

TABLEAU 7.2 – Résultats de segmentation sémantique sur le jeu de données ISPRS Potsdam à 12,5 cm/px (scores F_1 et exactitude globale).

Jeu de données	Méthode	Routes	Bâtiments	Vég. basse	Arbres	Véhicules	Exactitude
Validation 12,5 cm/px	SegNet RVB	92,4± 0,6	95,8± 1,9	85,8± 1,3	83,0± 2,1	95,7± 0,3	90,6± 0,6
Test 5 cm/px	SegNet IRRV	92,4	95,8	86,7	87,4	95,1	90,0
	FCN + CRF [50]	91,8	95,9	86,3	87,7	89,2	89,7

Potsdam, comme la vérité terrain initiale est constituée d'annotations pixelliques denses, nous avons régressé une boîte englobante pour chaque composante connexe en corrigeant manuellement les rares erreurs.

Nous suivons les pratiques habituelles en détection d'objet [15] et nous définissons un vrai positif comme une boîte englobante prédite dont l'intersection sur union (IsU) avec une boîte englobante de la vérité terrain est supérieure à 0,5. Si plusieurs prédictions existent pour le même véhicule, nous conservons celle avec la plus haute valeur d'IsU, le reste étant considéré comme des fausses alarmes. Sur le jeu de données NZAM/ONERA Christchurch, nous comparons nos résultats sur la même tuile que celle choisie par RANDRIANARIVO et al. [44], qui ont appliqué un *Discriminatively trained Model Mixture* (DtMM) contenant cinq modèles, un pour chaque orientation principale.

Pour évaluer l'effet de l'ouverture morphologique sur les performances en segmentation d'instance, nous détaillons dans le Tableau 7.4 la moyenne de l'IsU calculée sur les instances de véhicules et les scores précision/rappel finaux en fonction des divers prétraitements appliqués. Ceci montre que la simple ouverture morphologique et l'élimination des véhicules candidats dont la surface est inférieure à 100px permet de grandement améliorer les performances en éliminant les faux positifs. Ceci est particulièrement vrai pour le jeu de données

Tableau 7.3 – Résultats de segmentation sémantique sur NZAM/ONERA Christchurch (scores F_1 et exactitude globale).

	Arrière-plan	Bâtiments	Végétation	Véhicules	Exactitude
RVB	75,6± 8,9	91,7± 1,3	55,2± 11,6	61,9± 2,4	84,4± 2,6



Christchurch Christchurch Potsdam

Figure 7.6 – Exemples de détections sur Potsdam et Christchurch (vrais positifs en vert, faux positifs en rouge et vérité terrain en bleu).

NZAM/ONERA dans lequel les annotations grossières conduisent à des cartes sémantiques imprécises après inférence. Le processus complet atteint un score IsU de plus de 74% sur Potsdam et plus de 70% sur Christchurch.

Finalement, nous indiquons les résultats de détection dans le Tableau 7.5. Sur NZAM/ONERA Christchurch, notre méthode *segment-before-detect* obtient des résultats significativement supérieurs aux approches par mélange de modèles et HOG+SVM. Bien qu'aucune autre méthode de détection de véhicules n'ait été appliquée sur Potsdam, nous indiquons également nos scores de précision/rappel sur ce jeu de données. Des exemples qualitatifs de détection sont présentés dans la Figure 7.6.

Sur Christchurch, où les annotations sont grossières, le score F_1 atteint 0,81, tandis qu'il dépasse 0,87 sur Potsdam. En comparaison, l'approche classique par modèles déformables [43] n'atteint qu'un score F_1 de seulement 0,74. Notons par ailleurs que nous indiquons nos résultats pour lesquels un vrai positif est défini comme ayant un score IsU d'au moins 0,5. Dans la littérature, à des résolutions similaires (< 30 cm), la majorité des travaux considèrent qu'un vrai positif sur des véhicules de cette taille correspond à un seuil de 0,25. Ainsi, sur un jeu de données similaire, Tang et al. [56] obtiennent un score F_1 de 0,83 en utilisant le réseau de détection d'objets Faster-RCNN [47]. De façon postérieure à nos travaux, Van Etten [59] a introduit un jeu de données de détection de véhicules à des résolutions équivalentes (entre 10 cm et 1 m) et a adapté le réseau YOLO [46] pour l'imagerie aérienne. Il parvient ainsi obtenir un score F_1 d'environ 0,90 et à prédire le nombre de véhicules avec une erreur relative de 5%. Toutefois, cette approche ne prédit d'une part que des boîtes englobantes et le seuil considéré pour le score F_1 est fixé à 0,25, ce qui est bien plus permissif que celui que nous avons choisi. Dans l'ensemble, l'approche de segmentation avant détection paraît très compétitive avec les méthodes de détection pure, et permet d'extraire les instances d'objet bien plus finement en inférant des formes plutôt que des boîtes englobantes.

Christchurch est un jeu de données plus complexe pour deux raisons. Tout d'abord, la densité de véhicules y est nettement plus élevée que pour Potsdam, la ville comprenant de nombreux véhicules resserrés sur des surfaces réduites. Ensuite, les annotations grossières empêchent le FCN de prédire correctement les frontières des objets constituant la scène, ce qui conduit à des masques de véhicules imprécis (cf. Figure 7.5, l'IsU moyen sur Christchurch atteint 66,6% contre plus de 80% pour Potsdam). Cette combinaison rend le problème d'extraction des instances de véhicules plus difficile, mais reste toutefois à la portée de notre

Tableau 7.4 – Segmentation d'instance et détection de véhicules pour différents prétraitements morphologiques (IsU moyen, précision et rappel).

Jeu de données	prétraitement	IsU moyen	Précision	Rappel
NZAM/ONERA Christchurch	∅	60,0%	0,597	0,797
	Ouverture	69,8%	0,817	0,791
	Ouverture + retrait des petits objets	70,7%	0,833	0,791
ISPRS Potsdam	∅	70,1%	0,748	0,842
	Ouverture	73,3%	0,866	0,842
	Ouverture + retrait des petits objets	74,2%	0,907	0,841

Tableau 7.5 – Résultats de détection de véhicules sur Potsdam et Christchurch.

Jeu de données	Méthode	Précision	Rappel
NZAM/ONERA Christchurch	HOG+SVM [37]	0,402	0,398
	DtMM (5 modèles) [44]	0,743	0,737
	Segment-before-detect	**0,833**	**0,791**
ISPRS Potsdam	*Segment-before-detect*	0,907	0,841

approche. Cependant, les boîtes englobantes obtenues tendent à couvrir plusieurs véhicules. Les résultats de segmentation sémantique restent élevés en dépit de l'apprentissage sur des annotations prévues pour la détection d'objet. Ainsi, la segmentation peut être utilisée comme étape intermédiaire pour des tâches de détection, pour lesquelles l'état de l'art nécessite des approches sophistiquées et complexes. L'étape d'extraction de composantes connexes pourrait en outre être améliorée en utilisant une extraction de boîte englobante plus robuste, soit par des approches morphologiques comme un *watershed* appliqué aux cartes de probabilités [6, 4], soit en intégrant la prédiction d'instance au sein du réseau [13, 21].

Estimation de la densité de véhicules

Une fois les véhicules extraits des images, une tâche simple à réaliser consiste à estimer leur nombre sur une zone donnée. Nous divisons les deux jeux de données en cellule de 1000×1000px (soit $125 \times 125\,\mathrm{m}^2$) et nous comparons le nombre de véhicules prédits au nombre de véhicules contenus dans la vérité terrain :

$$\mathcal{E}_{relative} = \frac{|\text{nombre de véhicules prédits} - \text{nombre réel de véhicules}|}{\text{nombre réel de véhicules}} \,. \tag{7.1}$$

Les résultats sont moyennés et arrondis à l'entier le plus proche pour chaque jeu de données et détaillés dans le Tableau 7.6. Sur Potsdam comme Christchurch, les estimations sont justes avec moins de 10% d'erreur (±5 véhicules en moyenne). Les estimations sur Christchurch ont une erreur légèrement supérieure en raison de la segmentation et des détections moins précises.

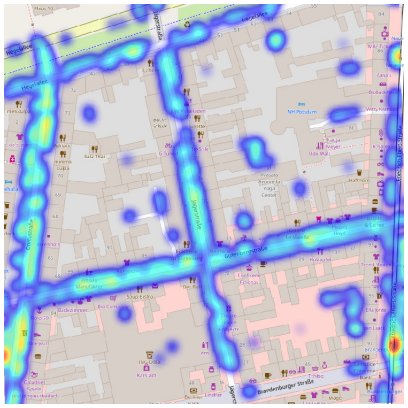
Une fois cette densité calculée, il est possible de réduire la taille des cellules et de produire des cartes de densité de présence de véhicules, comme illustrées dans les Figures 7.7 et 7.8. Ce type de cartes de densité peut ensuite être intégré à des SIG comme OpenStreetMap pour identifier automatiquement des routes encombrées, des parkings [27], etc.

Tableau 7.6 – Erreur moyenne d'estimation du nombre de véhicules par cellule de $125\,m^2 \times 125\,m^2$.

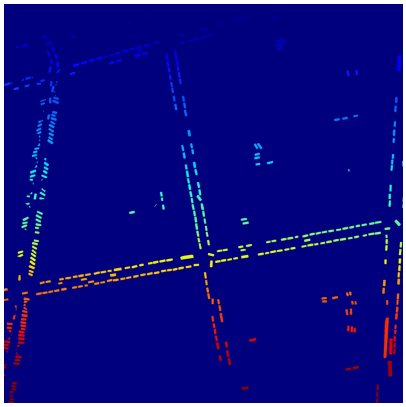
Jeu de données	ISPRS Potsdam	NZAM/ONERA Christchurch
Erreur absolue (erreur moyenne/total dans la vérité terrain)	3/52	6/66
Erreur relative	7,9%	9,1%



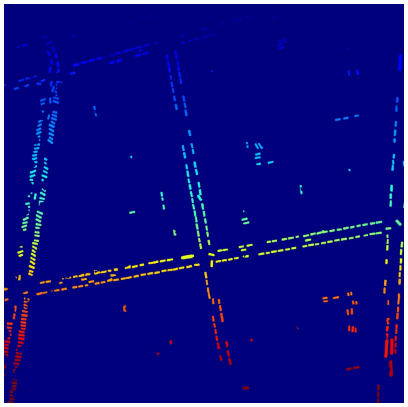
(a) Image RVB



(b) Densité de véhicules (Potsdam)



(c) Vérité terrain (Potsdam)



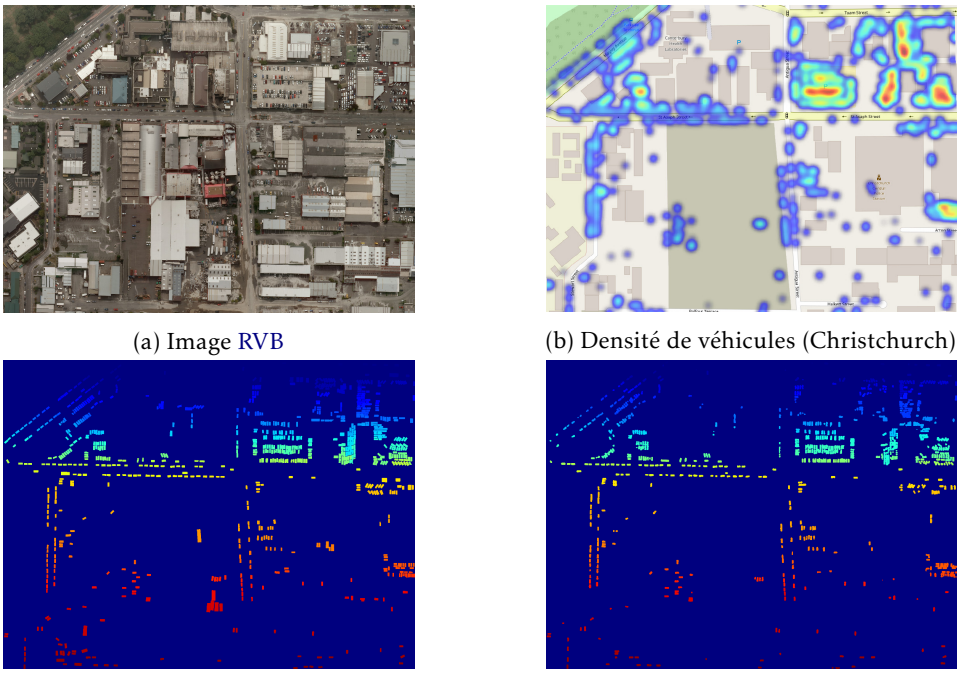
(d) Véhicules prédits (Potsdam)

Figure 7.7 – Visualisation des véhicules présents sur une des tuiles du jeu de données ISPRS Potsdam.

7.1.4 Classification de véhicules

Les véhicules étant désormais localisés et segmentés dans l'image, il s'agit enfin d'en identifier le type. À cette fin, nous comparons trois architectures de CNN à la complexité croissante : LeNet [32], AlexNet [29] et VGG-16 [51].

LeNet-5 est un petit CNN que nous entraînons à partir d'une initialisation aléatoire sur les véhicules de VEDAI en utilisant des images de dimensions 32×32. AlexNet et VGG-16 sont deux réseaux ayant gagné la compétition ILSVRC en 2012 et 2014. Des expériences préliminaires montrent que l'utilisation des poids de ces réseaux préentraînés sur ImageNet permet d'augmenter de 10% l'exactitude des modèles, ce qui est cohérent avec la littérature [38, 40]. Par conséquent, ces CNN seront simplement spécialisés sur des images de véhicules à résolution 224×224 et 227×227. Ces tailles d'images permettent de conserver les poids préentraînés des couches entièrement connectées. En pratique, les véhicules dans des images aériennes auront une taille d'environ 25×25. Nous extrayons donc des imagettes centrées sur ces véhicules incluant un contexte spatial de 16 px dans toutes les directions, cette approche ayant donné les meilleurs résultats. Un contexte plus large risquerait d'inclure d'autres véhicules dans la même imagette tandis qu'un contexte plus restreint diminue la quantité d'information disponible pour la classification. Les imagettes sont ensuites redimensionnées



(a) Image RVB (b) Densité de véhicules (Christchurch)

(c) Vérité terrain (Christchurch) (d) Véhicules prédits (Christchurch)

FIGURE 7.8 – Visualisation des véhicules présents sur une des tuiles du jeu de données NZAM/ONERA Christchurch.

par interpolation bilinéaire selon leur plus grande dimension pour correspondre à la taille attendue par le CNN, la plus petite dimension étant remplie par du bruit blanc.

Les modèles sont entraînés (ou spécialisés) durant 20 passes sur le jeu de données à l'aide d'une descente de gradient stochastique avec moment. Nous utilisons des mini-lots de 128 échantillons pour AlexNet et LeNet et de 32 pour VGG-16 compte-tenu de l'espace mémoire requis. Le taux d'apprentissage est initialement fixé à 0,01 et est divisé par 10 après 75% de l'entraînement. Pour les modèles soumis au *fine-tuning*, nous ré-entraînons le réseau dans son intégralité à l'exception de la dernière couche, qui est initialisée aléatoirement et est entraînée avec un taux d'apprentissage 10 fois supérieur. Nous appliquons également du *Dropout* [54] aux couches entièrement connectées pour limiter le surapprentissage.

Sans surprise, les performances des CNN sur VEDAI sont proportionnelles à leurs résultats sur ImageNet, comme montré dans le Tableau 7.7. Toutefois, le modèle le plus complexe (VGG-16) n'améliore que légèrement les résultats de classification comparé à l'important surcoût calculatoire qu'il engendre. En pratique, il pourrait être possible d'utiliser n'importe quel CNN, incluant les ResNet [20]. Toutefois, nous nous contentons du modèle AlexNet qui offre un rapport performance/temps d'exécution acceptable.

Le Tableau 7.8 détaille les résultats de classification de véhicules sur VEDAI en utilisant plusieurs stratégies de prétraitement des données. L'augmentation de données par transformations géométriques (noté *AD*) permet d'améliorer la robustesse et la généralisabilité du modèle. Le réalignement *R* permet également d'améliorer les résultats et de rendre le réseau plus robuste en éliminant la nécessité d'apprendre une classification équivariante par rotation. La combinaison de ces deux stratégies permettant d'obtenir les meilleurs résultats, nous les appliquons donc toutes deux sur les jeux de données ISPRS Potsdam et NZAM/ONERA Christchurch.

Transfert de connaissances pour la classification de véhicules

À ce stade, nous disposons d'un détecteur de véhicules efficace pour Potsdam et Christchurch et d'un classifieur entraîné sur VEDAI. Le Tableau 7.9 détaille donc les résultats

Tableau 7.7 – Résultats de classification de plusieurs CNN sur VEDAI (en %). OA = *Overall Accuracy* (exactitude globale).

Modèle	Voiture	Camion	Bateau	Tracteur	Camping-car	Van	Pick-up	Avion	Autres	OA	Temps (ms)
LeNet	74,3	54,4	31,0	61,1	85,9	38,3	7,7	13,0	47,5	66,3± 1,7	**2,1**
AlexNet	**91,0**	84,8	81,4	83,3	98,0	**71,1**	85,2	91,4	**77,8**	87,5± 1,5	5,7
VGG-16	90,2	**86,9**	**86,9**	**86,5**	**99,6**	**71,1**	**91,4**	**100,0**	77,2	**89,7**± 1,5	31,7

Tableau 7.8 – Résultats de classification d'AlexNet sur VEDAI avec plusieurs prétraitements (en %). OA = *Overall Accuracy* (exactitude globale), AA = *Average Accuracy* (exactitude moyenne).

prétraitement	Voiture	Camion	Bateau	Tracteur	Camping-car	Van	Pick-up	Avion	Autres	OA	AA
∅	90,4	66,7	80,4	**89,5**	96,6	63,3	78,7	92,6	75,0	83,9 ± 2.7	81,5± 1,9
AD	88,2	82,2	78,4	82,5	**97,4**	63,3	85,1	66,7	73,3	85,6± 1,4	77,3± 8,7
R	87,9	71,1	86,3	84,2	**97,4**	73,3	**87,2**	**100,0**	75,0	86,1± 0,9	84,7± 1,7
AD + R	**91,4**	**85,6**	**88,2**	87,6	**97,4**	**70,0**	**87,2**	**100,0**	**81,7**	**89,0**± 0,5	**87,7**± 1,5

AD = augmentation de données, R = re-normalisation.

de classification du CNN entraîné sur VEDAI et appliqué aux véhicules de Potsdam et Christchurch. Les résultats sont agrégés par validation croisée sur les mêmes sous-divisions des jeux de données que pour les résultats en segmentation sémantique. Les voitures étant majoritaires dans les jeux de données considérés, nous indiquons également les résultats qui seraient obtenus par une heuristique de référence correspondant à un classifieur renvoyant systématiquement "voiture". Ce classifieur constant serait correct 94% du temps, mais serait incapable de prédire autre chose qu'une voiture. L'exactitude globale du modèle serait donc excellente mais son exactitude moyenne catastrophique. Les CNN parviennent quant à eux à prédire correctement plusieurs types de véhicules, augmentant significativement l'exactitude moyenne tout en maintenant une exactitude globale compétitive. Des exemples qualitatifs de bonnes segmentations mais mauvaises classifications et de bonnes segmentations et classifications sont illustrés dans les Figures 7.9 et 7.10.

L'exactitude moyenne est plus basse sur Potsdam que sur VEDAI compte-tenu du fort déséquilibre entre classes et de la sensibilité numérique des résultats. En effet, chaque sous-division entraînement/test de la validation croisée contient environ 15 exemples de camions et de pick-ups. Cependant, le modèle est entraîné sur VEDAI, dont la répartition entre classes n'est pas autant déséquilibrée. Par conséquent, le modèle possède un biais qui ne se transfère pas à Potsdam. En outre, les capteurs utilisés pour VEDAI, Potsdam et Christchurch sont différents, tout comme les environnements considérés (urbain pour Potsdam et Christchurch, rural pour VEDAI).

Les variations dues aux capteurs ont été corrigées en renormalisant la colorimétrie des images de Potsdam et Christchurch. Pour cela, nous estimons les moments statistiques des pixels sur VEDAI et nous les appliquons à Potsdam et Christchurch :

$$\mathrm{I}_{test} := \frac{\mathrm{I}_{test} - m_{test}}{\sigma^2_{test}} \cdot \sigma^2_{VEDAI} + m_{VEDAI} \tag{7.2}$$

avec m la valeur du pixel moyen dans le jeu de données, σ l'écart-type et I l'image à transformer. Cette opération est appliquée sur chaque canal.

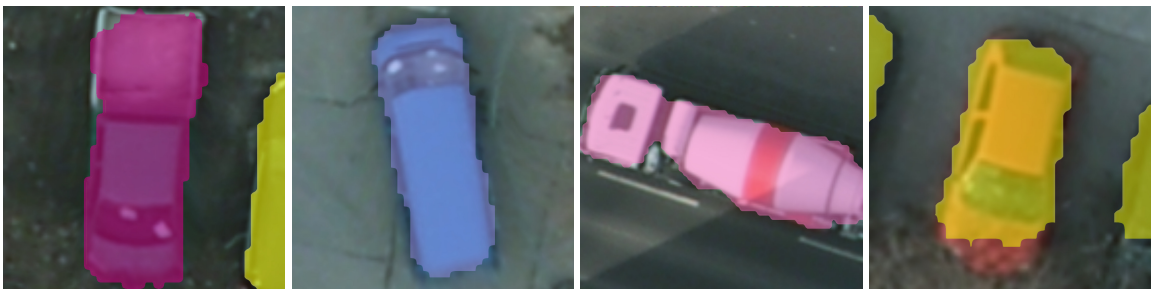
Toutefois, les différences d'apparence de l'environnement et surtout des types des véhicules diminuent tout de même les performances. Notamment, les voitures, camions et pick-ups sur Christchurch sont plus proches des marques américaines présentes dans VEDAI que les véhicules européens de Potsdam. Ces variations environnementales et d'apparence des objets peuvent faire sortir le classifieur de sa plage de fonctionnement nominal.

Une régularisation adéquate ou un entraînement sur un jeu de données de véhicules plus varié pourrait permettre de limiter ces effets négatifs. De manière générale, ce type de



(a) Van (camion prédit) (b) Van (camion prédit) (c) Voiture (van prédit) (d) Pick-up (van prédit)

FIGURE 7.9 – Segmentations réussies mais mauvaises classifications sur Potsdam.



(a) Pick-up (b) Van (c) Camion (d) Voiture

FIGURE 7.10 – Segmentations et classifications réussies sur Potsdam.

transfert de connaissances est lié au problème d'adaptation de domaine non-supervisée [57, 12], qui est un sujet de recherche important en télédétection.

Finalement, nous avons donc montré qu'il était possible de récupérer *a posteriori* la structure des objets segmentés par un FCN directement depuis la classification pixel à pixel. Notamment, la méthode *segment-before-detect* permet non seulement de détecter les véhicules dans des images aériennes, mais également d'identifier précisément leur forme et leur type. En particulier, cette méthode s'applique y compris dans le cas d'annotations imprécises, comme les boîtes englobantes de NZAM/ONERA Christchurch. Toutefois, l'extraction des instances d'objet nécessite de faire appel à un processus *ad hoc* permettant de regrouper les pixels appartenant à un même objet. Comme nous l'avons vu jusqu'ici, les modèles entièrement convolutifs sont optimisés pour minimiser une fonction de coût de classification pixel à pixel. Une telle fonction de coût ne permet pas de modéliser efficacement les dépendances spatiales pouvant exister entre les pixels, notamment leur appartenance à un même objet. Il paraît donc pertinent d'étudier comment rendre compte de ces relations spatiales dans l'optimisation des réseaux.

Tableau 7.9 – Résultats de classification de véhicules sur les vérités terrain augmentées de Potsdam et Christchurch. OA = *Overall Accuracy* (exactitude globale), AA = *Average Accuracy* (exactitude moyenne).

Jeu de données	Classifieur	Voiture	Van	Camion	Pick-up	OA	AA
Potsdam	Voitures seulement	100%	0%	0%	0%	94%	25%
	AlexNet	**98%**	**66%**	67%	0%	**95%**	58%
	VGG-16	92%	66%	**75%**	**33%**	89%	**67%**
Christchurch	Voitures seulement	100%	0%	0%	0%	94%	25%
	AlexNet	94%	40%	**67%**	**89%**	93%	73%
	VGG-16	**97%**	**80%**	**67%**	78%	**96%**	**80%**

7.2 Segmentation sémantique par régression des cartes de distances

7.2.1 Annotations sémantiques et cartes de distances

Comme nous l'avons vu dans la section précédente, il est possible de reconstruire *a posteriori* la structure des objets dans les cartes sémantiques prédites par un FCN. Toutefois, la littérature relaie régulièrement des problèmes de frontières inter-classes imprécises ou de segmentations bruitées, nécessitant de faire appel à des régularisations a posteriori pour lisser les segmentations [63, 9] ou à des post-traitements *ad hoc* comme la méthode *segment-before-detect*.

La communauté s'est ainsi penchée sur différents post-traitements pour améliorer la netteté des contours et contraindre les segmentations à respecter la même topologie que la vérité terrain. Bien souvent, il s'agit de modèles graphiques ajoutés en fin de réseau [33] ou faisant appel à des connaissances a priori [30, 5]. La segmentation d'instance en particulier s'intéresse à combiner des approches de localisation géométrique avec des approches de sémantisation [21, 13].

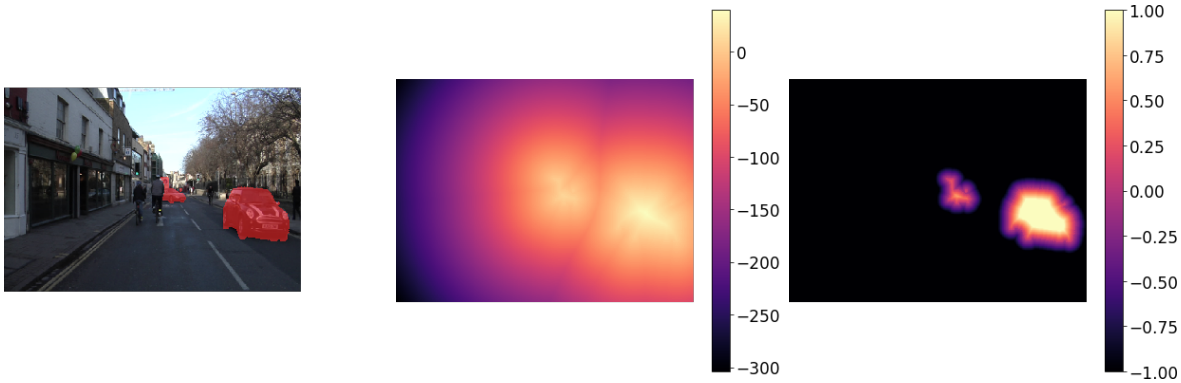
Nous présentons ici une approche directe consistant en une régularisation implicite intégrée dans la représentation de la vérité terrain. En effet, nous proposons d'utiliser l'estimation des cartes de distance issues des masques de segmentation comme tâche auxiliaire. Les cartes de distance indiquent non seulement l'appartenance d'un pixel à une classe donnée, mais également sa proximité spatiale vis-à-vis des autres classes d'intérêt et contient donc une information plus riche concernant la structure spatiale des données. Cette approche s'inscrit dans la veine de travaux sur l'utilisation de primitives géométriques pour régulariser la segmentation sémantique, comme la prédiction de l'orientation des objets [58] ou de la position de leur centre de masse [17]

De fait, en modifiant de façon minimale des réseaux de segmentation existants, nous parvenons à obtenir des segmentation plus régulières sans post-traitement ou connaissance a priori.

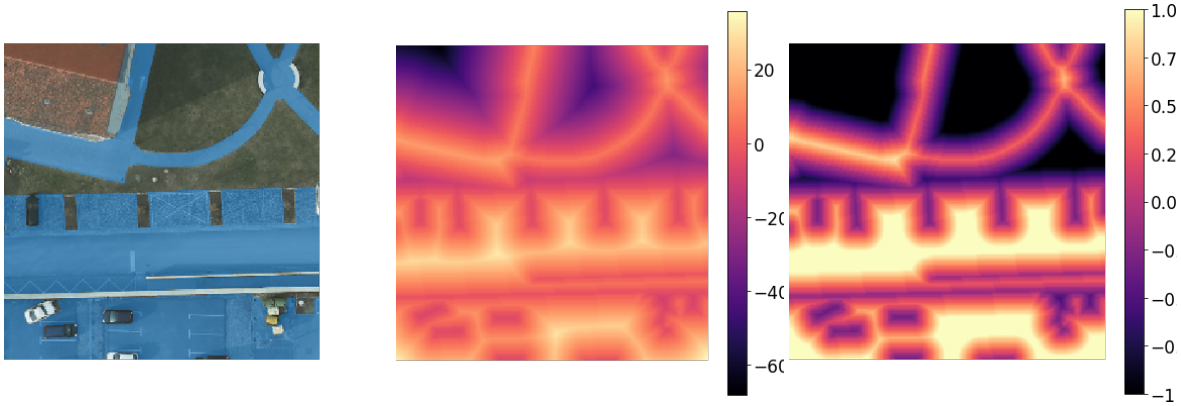
Nous validons notre méthode sur plusieurs architectures de réseaux convolutifs profonds et sur plusieurs applications en compréhension de scène urbaine, en segmentation d'images 2,5D et en observation de la Terre.

Régularisation par régression des cartes de distances

Le passage en carte de distances transforme un masque binaire en une représentation équivalente à valeurs continues. En l'occurence, nous travaillons avec des cartes de distances signées tronquées puis renormalisées dans $[-1, +1]$. Ces représentations des annotations sont illustrées dans la Figure 7.11. Nous émettons l'hypothèse que cette représentation permet toutefois d'accéder plus directement à la structure spatiale des données, notamment car elle contient pour un pixel sa distance spatiale relative à toutes les classes d'intérêt. Cette



(a) Masque binaire. (b) Carte de distance signée (c) CDS tronquée et normalisée.



(d) Masque binaire. (e) Carte de distance signée (f) CDS tronquée et normalisée.

Figure 7.11 – Différentes représentations de segments annotés.

représentation explicite ainsi mieux la géométrie de la scène que les masques binaires utilisés pour la classification. Nous montrons que la régression des cartes de distances signées (CDS) en tâche auxiliaire d'un réseau de segmentation sémantique a des résultats bénéfiques sur la segmentation finale.

Transformée de distances

La transformée de distances (ou carte de distances) d'une image binaire assigne à chaque pixel du maillage sa distance au point positif du masque le plus proche. La distance peut être calculée en utilisant différentes métriques, comme la distance Manhattan ou la distance Euclidienne. Pour les éléments appartenant au masque positif (premier-plan), la distance est conventionnellement 0. Par exemple, la transformée $\mathcal{D}$ de distances euclidienne transforme une image I de dimensions $\mathrm{M} \times \mathrm{N}$ de masque positif I^+ en une carte de distance $\mathcal{D}(\mathrm{I})$ obtenue par :

$$\forall i, j \in \mathrm{M} \times \mathrm{N}, \quad \mathcal{D}(\mathrm{I})[i,j] = \min_{\mathrm{I}_{i',j'} \in \mathrm{I}^+} (\| \mathrm{I}[i,j] - \mathrm{I}[i',j'] \|) \quad . \tag{7.3}$$

En pratique, nous utilisons la transformée de distances signée [60], qui associe à chaque pixel du premier plan sa distance positive au pixel de l'arrière-plan le plus proche et à chaque pixel de l'arrière-plan l'opposé de sa distance au pixel du premier plan le plus proche. Formellement, il s'agit de la transformée $\mathcal{D}_s$ qui associe à I son image :

$$\forall i, j \in \mathrm{M} \times \mathrm{N}, \quad \mathcal{D}_s(\mathrm{I})[i,j] = \begin{cases} +\min_{\mathrm{I}_{i',j'} \in \mathrm{I}^-} (\| \mathrm{I}[i,j] - \mathrm{I}[i',j'] \|), & \text{si } \mathrm{I}[i,j] \in \mathrm{I}^+, \\ -\min_{\mathrm{I}_{i',j'} \in \mathrm{I}^+} (\| \mathrm{I}[i,j] - \mathrm{I}[i',j'] \|), & \text{si } \mathrm{I}[i,j] \notin \mathrm{I}^+. \end{cases} \tag{7.4}$$

Les annotations de segmentation sémantique correspondent en pratique à un masque binaire par classe. Il est donc possible de convertir ces annotations en leur contrepartie sous

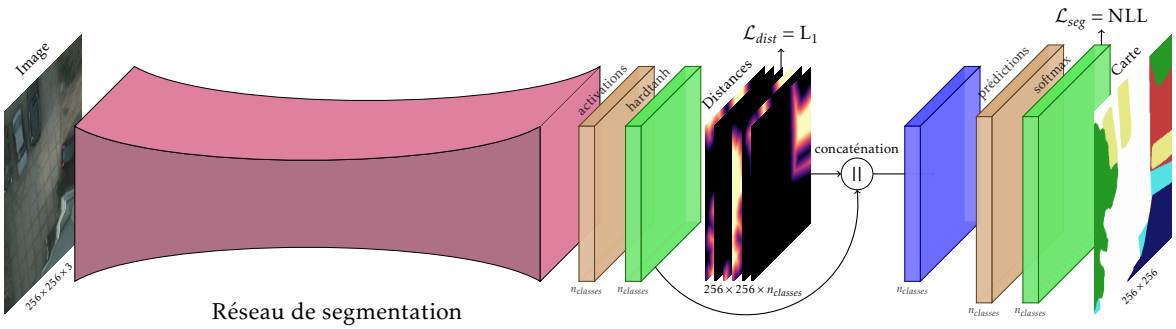


FIGURE 7.12 – Apprentissage multitâche (classification pixel à pixel et régression des cartes de distances). Les couches convolutives sont en bleu, les activations non-linéaires en vert et les cartes d'activation en marron.

forme de CDS. Il est important de souligner qu'aucune information n'est perdue dans ce procédé, les masques binaires pouvant être retrouvées par simple seuillage des CDS. Nous appliquons donc la transformée de distances signée aux annotations en utilisant l'algorithme exact en temps linéaire de MAURER, QI et RAGHAVAN [36].

Pour limiter des effets indésirables lorsque les pixels sont assez éloignés des autres objets qui sortent alors du champ réceptif du réseau, nous ajoutons une saturation aux distances calculées . En particulier, nous divisons la CDS par un facteur d'échelle puis nous normalisons la CDS dans $[-1,+1]$ en lui appliquant une transformation hardtanh. Ces différentes représentations sont illustrées dans la Figure 7.11.

7.2.2 Apprentissage multitâche

La régression directe des CDS ne permet pas d'obtenir de meilleurs résultats de segmentation que la classification dense pixel à pixel usuelle. De fait, nous proposons donc d'utiliser une stratégie d'apprentissage multitâche dans laquelle le réseau est optimisé à la fois sur la classification des pixels et sur la régression des CDS.

En particulier, nous modifions l'architecture du réseau pour, dans un premier temps, effectuer la régression des CDS ; puis nous ajoutons une couche convolutive additionnelle pour fusionner les activations de la dernière couche avec les CDS prédites afin de réaliser la classification finale. Le réseau est ainsi entraîné en multitâche, la régression des CDS étant utilisée comme tâche intermédiaire avant la classification.

L'altération du réseau se résume comme suit. La dernière couche, habituellement suivie d'un *softmax* est ici utilisée comme couche de régression des CDS. Les distances étant normalisées entre −1 et 1, la fonction *hardtanh* est utilisée comme activation non-linéaire. Puis nous concaténons les activations de la couche précédente aux CDS ainsi prédites pour alimenter une couche convolutive additionnelle suivie d'un *softmax* qui réalise la classification pixellique. L'architecture complète est illustrée dans la Figure 7.12. Par souci d'équité dans nos expériences, les modèles de référence présentés par la suite incluent cette même couche de convolution additionnelle afin que les réseaux originaux et modifiés possèdent le même nombre de paramètres optimisables.

Les fonctions de coût utilisées dans ces travaux sont la log-vraisemblance négative (NLL) sous forme d'entropie croisée pour la classification et la distance L_1 pour la régression. En notant respectivement $Z_{seg}, Z_{dist}, Y_{seg}, Y_{dist}$ la classification après *softmax*, la carte de distances prédite, les annotations de vérité terrain et la carte de distances réelle, la fonction de coût à minimiser est :

$$\mathcal{L}_{totale} = \mathrm{NLL}(Z_{seg}, Y_{seg}) + \lambda \cdot L_1(Z_{dist}, Y_{dist}) \tag{7.5}$$

où λ est un hyperparamètre contrôlant l'amplitude de la régularisation.

7.2.3 Validation expérimentale

Afin de pouvoir mesurer l'effet de la régression des cartes de distance, nous entraînons des réseaux avec l'architecture SegNet [3] ou PSPNet [62] de référence, soit en régression pure, soit en classification pure.

L'architecture encodeur-décodeur SegNet [3] a déjà été présentée au Chapitre 3. PSPNet [62] est une architecture entièrement convolutive ayant établi un nouvel état de l'art sur plusieurs jeux de données de segmentation sémantique [11, 15]. Elle dérive du modèle ResNet [20] et utilise un module de concaténation en pyramide d'activations pour prendre en compte plusieurs niveaux de contexte spatial. Dans notre cas, nous utilisons une version réduite de PSPNet conçue sur l'architecture ResNet-101. ResNet-101 produit des cartes d'activation à résolution 1:32 qui sont suréchantillonnées par déconvolution.

Jeux de données

Tableau 7.10 – Résultats de validation croisée sur les jeux de données ISPRS (multitâche). Les valeurs indiquées représentent le taux global de bonne classification et le score F_1 pour chaque classe.

Méthode	Ville	Exac.	Routes	Bâtiments	Vég. basse	Arbres	Véhicules
SegNet* (régression)	Vaihingen	89,49	91,03	95,60	81,23	88,31	0,00
SegNet* (classification)		90,00	91,98	95,53	80,91	88,07	87,94
SegNet* (multitâche)		**90,43**	**92,46**	**95,99**	**81,30**	**88,34**	**88,16**
SegNet* (classification)	Potsdam	91,85	94,12	96,09	88,48	85,44	96,62
SegNet* (multitâche)		**92,22**	**94,33**	**96,52**	**88,55**	**86,55**	**96,79**

Nous validons notre approche sur plusieurs jeux de données afin de démontrer sa capacité à généraliser dans des contextes de segmentation mono et multiclasse sur plusieurs types d'images.

ISPRS 2D Semantic Labeling Le jeu de données ISPRS 2D Semantic Labeling [48] est constitué des scènes Potsdam et Vaihingen, déjà présentées aux chapitres précédents et détaillés dans l'Annexe A.1.1. L'évaluation est réalisée par validation croisée en divisant les jeux de données en trois plis.

INRIA Aerial Image Labeling Benchmark Le jeu de données INRIA Aerial Image Labeling [34] contient 360 images RVB de taille 5000×5000px à une résolution de 30cm/px, couvrant 10 agglomérations de divers points du globe. La moitié des villes sont utilisées pour l'apprentissage et associées à des annotations publiques d'empreintes de bâtiments. Le reste du jeu de données est utilisé pour l'évaluation. Plus d'informations sont données dans l'Annexe A.1.4.

CamVid Le jeu de données CamVid [7] comporte 701 images extraites de plusieurs vidéos filmées par une caméra embarquée dans une voiture, avec une résolution de 360×480px. Nous utilisons la division du jeu de données de référence [3], c'est-à-dire 367 images d'apprentissage, 101 images de validation et 233 images de test. Les annotations recouvrent 11 classes d'intérêt telles que "bâtiment", "piéton", "voiture" ou encore "trottoir". Plus de détails sont donnés dans l'Annexe A.2.1.

SUN RGB-D Le jeu de données SUN RGB-D [53] contient 10 335 images RVB accompagnées d'une carte de profondeur. Ces images ont été annotées sur 37 classes d'intérêt comportant

le mobilier, les murs, le sol... Il vise principalement à évaluer les capacités d'interprétation d'images dans un cadre de navigation robotique en intérieur, avec des objets situés à moins de 10 m. Plus d'informations sont données dans l'Annexe A.2.2.

Data Fusion Contest 2015 Le jeu de données Data Fusion Contest 2015 [8] comporte 7 images aériennes de dimensions 10 000 px × 10 000 px à une résolution de 5 cm/px, acquises sur la ville de Zeebruges (Belgique). 8 classes d'intérêt (les 6 classes des jeux de données ISPRS ainsi que l'eau et les bateaux) sont annotées. Nous conservons deux images pour le test, une image pour la validation et le reste pour l'entraînement. Plus de détails sur le jeu de données se trouvent dans l'Annexe A.1.2.

Protocole expérimental

Les architectures SegNet et PSPNet-101 sont entraînées et déployées de la façon suivante. SegNet est entraîné pendant 50 000 itérations sur des *mini-batches* de 10 images. L'optimisation se fait par descente de gradient stochastique avec un taux d'apprentissage de 0,01, divisé par 10 après 25 000 et 45 000 itérations. Les poids de l'encodeur sont initialisés avec ceux de VGG-16 [51] préentraîné sur ImageNet. Les poids du décodeur sont initialisés aléatoirement en utilisant la stratégie de He et al. [19]. Pour le jeu de données multimodal SUN RGB-D, nous utilisons le modèle FuseNet [18], qui consiste en un SegNet à double entrée (cf. Chapitre 5). Sur les images aériennes, nous augmentons le nombre d'échantillons d'apprentissage en extrayant des images de 256 × 256 (384 × 384 pour le jeu de données INRIA Aerial Image) et en procédant aléatoirement à des symétries horizontales ou verticales. L'inférence est réalisée avec une fenêtre glissante de même dimension et un recouvrement de 75%.

PSPNet est entraîné sur CamVid pendant 750 000 itérations sur 10 images en parallèle par descente de gradient stochastique avec un taux d'apprentissage de 0,01, divisé par 10 après 500 000 itérations. Nous extrayons aléatoirement des imagettes de 224 × 224 et nous appliquons aléatoirement une symétrie horizontale. Suivant le protocole de [25], nous raffinons l'apprentissage en entraînant pendant 200 000 itérations sur les images à pleine résolution. Notre implémentation de PSPNet utilise les poids de ResNet-101 [20] préentraînés sur ImageNet pour l'initialisation, et n'utilise pas la fonction de coût auxiliaire présentée dans [62].

Finalement, nous compensons le déséquilibre des classes dans les jeux de données SUN RGB-D et CamVid en utilisant une pondération inversement proportionnelle à la fréquence médiane.

Toutes les expériences sont réalisées à l'aide de la bibliothèque PyTorch [41]. Les CDS sont calculées sur CPU à l'aide de la bibliothèque Scipy [26] et conservées en mémoire pour éviter les calculs redondants. Le calcul des CDS ralentit légèrement l'entraînement lors des premières itérations, avant leur mise en cache. Dans un cadre applicatif concret, une implémentation en ligne sur GPU [61] permettrait d'effectuer ces calculs en ligne rapidement sans nécessiter de surcoût mémoire.

Résultats

Dans les Tableaux 7.10 à 7.12 et 7.14, les modèles signalés par une étoile ("*") sont ceux proposés dans le cadre cette étude et implémentés par nos soins. Les autres modèles sont des références de l'état de l'art.

ISPRS Les résultats de validation croisée sur les jeux de données ISPRS Vaihingen et Potsdam sont détaillés dans le Tableau 7.10. Toutes les classes semblent bénéficier de la régression des cartes de distances. En particulier, les arbres sur Potsdam sont significativement mieux segmentés, la régression de la CDS contraignant le réseau à prendre en compte la convexité naturelle de l'objet en dépit de l'absence de feuilles. Deux exemples de segmentation sont

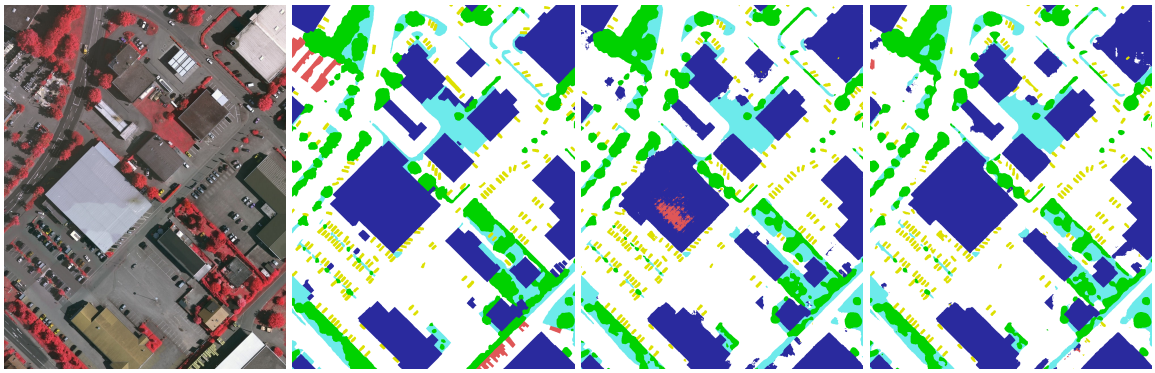


Image IRRV | Vérité terrain | SegNet (classification) | SegNet (multitâche)

FIGURE 7.13 – Extrait des résultats de segmentation sur le jeu de données ISPRS Vaihingen. Légende : blanc : routes, bleu : bâtiments, cyan : végétation basse, vert : arbres, jaune : véhicules, rouge : autre.

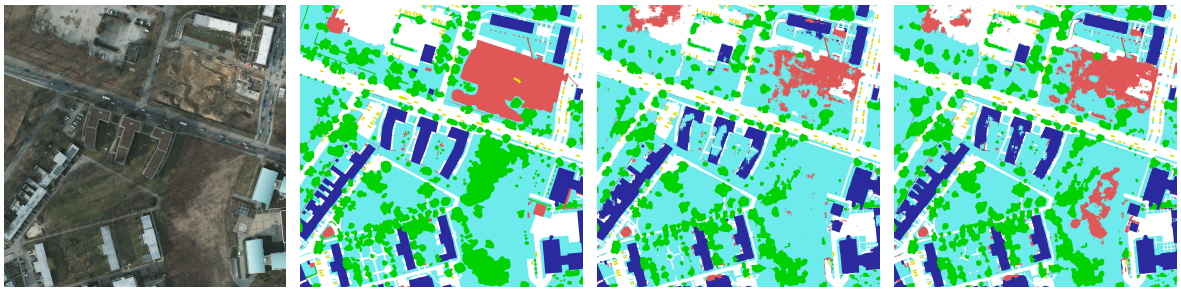


Image RVB | Vérité terrain | SegNet (classification) | SegNet (multitâche)

FIGURE 7.14 – Extrait des résultats de segmentation sur le jeu de données ISPRS Potsdam. Légende : blanc : routes, bleu : bâtiments, cyan : végétation basse, vert : arbres, jaune : véhicules, rouge : autre.

présentés en Figure 7.13 et Figure 7.14, dans lesquelles on peut voir que les bâtiments bénéficient grandement de l'approche multitâche (apparence plus lisse et moins de bruit de classification). Nous avons également testé l'approche par régression seule sur le jeu de données ISPRS Vaihingen avec des résultats mitigés. En effet, la plupart des classes bénéficient de ce traitement mais le réseau devient alors incapable de segmenter les véhicules, provoquant dans l'ensemble une baisse du taux de bonne classification.

INRIA Aerial Image Labeling Benchmark Les résultats sur le jeu de données INRIA *Aerial Image Labeling* sont détaillés dans le Tableau 7.11. L'utilisation de la régression sur les cartes de distances améliore significativement le score d'IsU. Comme illustré dans la Figure 7.15, les formes des bâtiments respectent mieux l'a priori polygonal et la connexité des objets. Les bâtiments qui étaient déjà détectés sont segmentés avec plus de régularité. Dans l'ensemble, nos résultats sont compétitifs avec les autres méthodes de la première année du comparatif [23], qui utilisent notamment une relaxation de l'indice de Jaccard comme fonction de coût.

SUN RGB-D Nous indiquons dans le Tableau 7.12 les résultats détaillés de segmentation sur le jeu de données SUN RGB-D. Le passage à un modèle multitâche améliore légèrement la précision moyenne et le taux moyen de bonne classification, contre une très faible diminution de l'IsU. Ces résultats montrent que l'utilisation de la régression sur les cartes de distances s'étend également à des architectures multimodales à double entrée. En outre, nos résultats sont comparables à ceux obtenus par QI et al. [42] utilisant un réseau de neurones convolutif sur le graphe 3D de la scène, qui utilise donc une information plus riche.

Tableau 7.11 – Performances en extraction de bâtiments sur le jeu de données INRIA *Aerial Image Labeling*.

Méthode	Bellingham		Bloomington		Innsbruck		San Francisco		East Tyrol		Global	
	IsU	Exac.	IsU	Exac.	IsU	Exac.	IsU	Exac.	IsU	Exac.	IsU	Exac.
AMLL [23]	67,14	96,64	65,43	96,73	72,27	96,66	**75,72**	**91,80**	74,67	97,70	**72,55**	**95,91**
NUS [23]	**70,74**	**97,00**	66,06	96,74	**73,17**	**96,75**	73,57	91,19	**76,06**	**97,81**	72,45	95,90
Raisa [23]	68,73	96,79	60,83	96,23	70,07	96,31	70,64	89,52	74,76	97,64	69,57	95,30
Inria [34]	56,11	95,37	50,40	95,27	61,03	95,37	61,38	87,00	62,51	96,61	59,31	93,93
SegNet* (classif.)	63,42	96,11	62,74	96,20	63,77	95,44	66,53	89,18	65,90	96,76	65,04	94,74
SegNet* (multi)	68,92	96,94	**68,12**	**97,00**	71,87	96,72	71,17	89,74	74,75	97,78	71,02	95,63

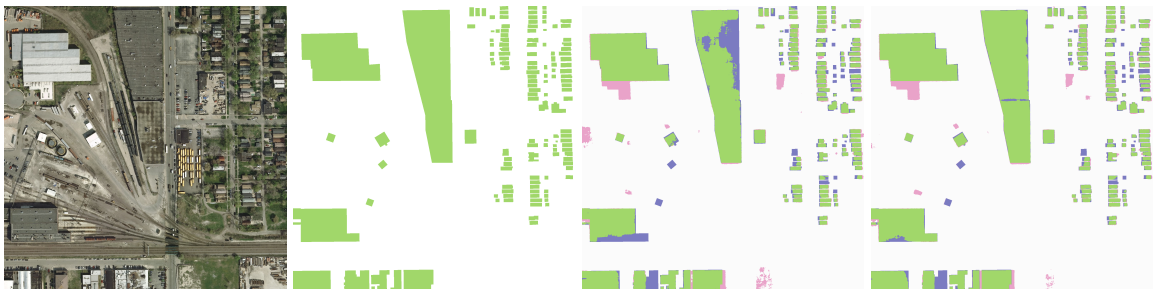


Image RVB | Vérité terrain | SegNet (classification) | SegNet (multitâche)

Figure 7.15 – Extrait des résultats de segmentation sur le jeu de données INRIA *Aerial Image Labeling*. Les pixels corrects sont en vert, les faux positifs en rose et les faux négatifs en bleu. L'approche multitâche capture mieux la structure spatiale des objets.

Data Fusion Contest 2015 Le Tableau 7.13 compile les résultats de segmentation sémantique obtenus en entraînant un SegNet avec et sans régression des CDS sur le jeu de données DFC 2015. À titre de comparaison, la meilleure approche de la compétition initiale est également indiquée [8]. Les gains quantitatifs sont similaires à ceux obtenus sur ISPRS Vaihingen et Potsdam. En effet, la plupart des classes bénéficient de la régularisation, en particulier la végétation ; la géométrie des annotations sur la végétation est nettement plus régulière que son aspect réel, à tendance chaotique et fractale. Dans l'ensemble, l'exactitude du modèle est améliorée de 0,64% dans le mode multitâche.

CamVid Les résultats sur le jeu de données CamVid sont détaillés dans le Tableau 7.14 avec notamment une comparaison à la méthode de Jégou et al. [25]. Plusieurs exemples qualitatifs sont présentés dans la Figure 7.16. Le passage de PSPNet au mode de fonctionnement multitâche permet d'améliorer l'IsU global de presque 2 points et améliore la majorité des classes, à l'exception du ciel et des routes. Ceci est notamment dû à la présence de pixels

Tableau 7.12 – Résultats sur le jeu de données SUN RGB-D (images de 224 × 224px).

Méthode	Exactitude	IsU	Précision
3D Graph CNN [42]	-	42,0	55,2
3D Graph CNN [42] (multiéchelle)	-	**43,1**	55,7
FuseNet* [18]	76,8	39,0	55,3
FuseNet* (multitâche)	**77,0**	38,9	**56,5**

Tableau 7.13 – Résultats de segmentation sémantique sur le jeu de données DFC 2015 (scores F_1 par classe et exactitude globale).

Méthode	Exac.	Routes	Bâtiments	Vég. basse	Arbres	Véhicules	Autre	Bateaux	Eau
AlexNet (par *patch*) [8]	83,32	79,10	75,60	78,00	79,50	50,80	63,40	44,80	98,20
SegNet (classification)	86,67	**84,05**	**82,21**	82,24	69,10	79,27	65,78	**56,80**	98,93
SegNet (multi-tâche)	**87,31**	84,04	81,71	**83,88**	**80,04**	**80,27**	**69,25**	50,83	**98,94**

Tableau 7.14 – Performances de segmentation sémantique sur le jeu de données CamVid.

Méthode	IsU	% classif.	Bâtiments	Arbres	Ciel	Voiture	Panneau	Route	Piéton	Barrière	Poteau	Trottoir	Cycliste
SegNet [3]	46,4	62,5	68,7	52,0	87,0	58,5	13,4	86,2	25,3	17,9	16,0	60,5	24,8
DeepLab-LFOV [9]	61,6	–	81,5	74,6	89,0	**82,2**	42,3	92,2	48,4	27,2	14,3	75,4	50,1
DenseNet56 [25]	58,9	88,9	77,6	72,0	92,4	73,2	31,8	92,8	37,9	26,2	32,6	79,9	31,1
DenseNet103 [25]	**66,9**	**91,5**	**83,0**	**77,3**	**93,0**	77,3	**43,9**	**94,5**	**59,6**	37,1	**37,8**	**82,2**	50,5
PSPNet* (classification)	60,3	89,3	74,7	64,1	89,0	71,8	36,6	90,8	44,5	38,5	25,4	77,4	50,3
PSPNet* (multitâche)	62,2	90,0	76,2	66,4	88,8	78,0	37,6	90,7	47,2	**40,1**	28,6	78,9	**51,2**

non annotés, nombreux aux frontières de ces classes, provoquant la génération de cartes de distances inexactes. Dans l'ensemble, nos résultats sont compétitifs avec les méthodes de l'état de l'art [25, 9].

Discussion

Afin de mieux comprendre l'influence de la pondération dans la fonction de coût, nous entraînons plusieurs modèles sur le jeu de données ISPRS Vaihingen avec différentes valeurs pour λ. Cela permet d'ajuster l'influence relative de la régression des CDS comparée à l'entropie croisée. Comme indiqué dans la Tableau 7.10, nous comparons la régression et la classification simultanée à chacune des tâches prises seules. En pratique, il s'avère que la régression des CDS seule obtient des performances de classification inférieures à celle de la classification. Cela correspond à un mode où $\lambda \to +\infty$, tandis que la classification seule correspond à $\lambda = 0$. Nous étudions donc les performances des modèles entraînés avec des valeurs intermédiaires de λ.

Comme illustré dans la Figure 7.17, incorporer la régression des cartes de distances permet d'améliorer les performances de classification significativement. Deux points de fonctionnement particulièrement avantageux apparaissent à $0,5$ et 2. Le premier souffre d'une variance élevée sur les différentes expériences, tandis que le second atteint une précision légèrement plus faible mais plus robuste. En pratique, toutes les valeurs de λ dans la plage considérée ont permis d'améliorer les performances du FCN, ce qui rassure donc sur la facilité à trouver une valeur adéquate pour cet hyperparamètre supplémentaire.

L'intégration de la régression des cartes de distances dans un cadre d'optimisation multi-tâche permet d'améliorer et de lisser la structure spatiale des segmentations prédites par le réseau. Le modèle est contraint d'apprendre la notion de proximité spatiale d'un pixel par rapport à des classes voisines. Par exemple, dans le cas des images aériennes, des arbres dont le feuillage tombe en hiver peuvent révéler le sol. La réponse spectrale des filtres correspond alors à un mélange de texture, bien que l'annotation recherchée corresponde à l'enveloppe convexe de l'arbre. La régression des cartes de distances permet d'orienter le réseau vers

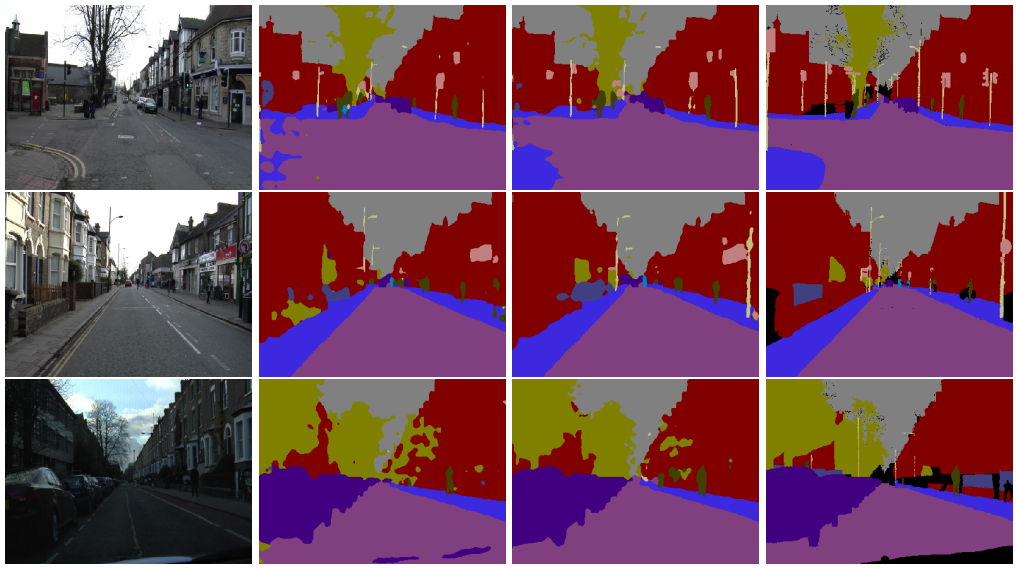
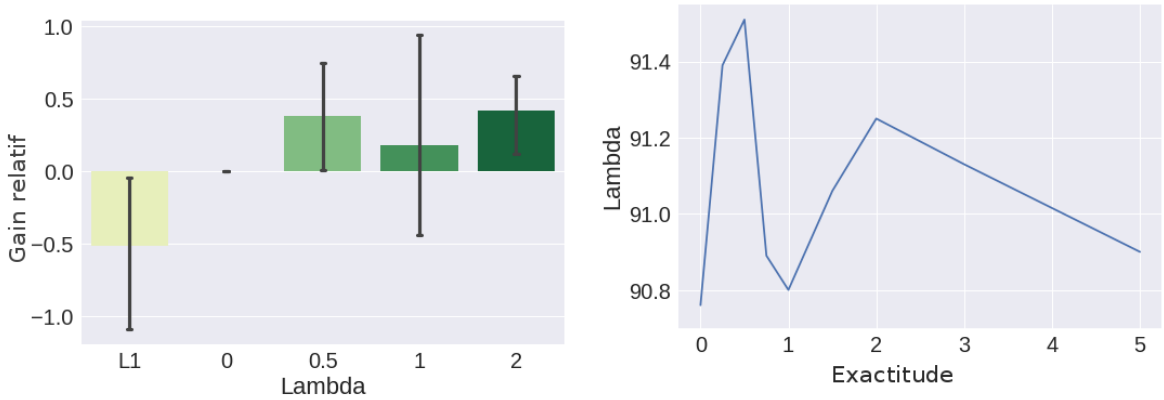


Figure 7.16 – Exemple de résultats de segmentation sémantique sur le jeu de données CamVid. De gauche à droite : image RVB, PSPNet (classification), PSPNet (multitâche), annotations.



(a) Gain en performance en fonction de λ.

(b) Évolution du score de classification en fonction de la valeur de λ.

Figure 7.17 – Exploration de plusieurs valeurs de λ sur le jeu de données ISPRS Vaihingen.

des recherches de structures géométriques, moins dépendantes de la radiométrie locale. En outre, cela permet de limiter la présence du bruit de classification poivre et sel qui est habituellement corrigé par des modèles graphiques a posteriori. Enfin, une piste de recherche intéressante pour les CDS réside dans la segmentation panoptique [28]. Cette tâche consiste à réaliser de façon conjointe la segmentation sémantique et la segmentation d'instances. En effet, les objets peuvent être définis par instance mais de nombreuses surfaces et zones (le ciel, la mer, les routes…) n'ont pas de notion explicite d'instance. Les cartes de distances ont l'avantage de pouvoir exprimer cette double notion par des surfaces de niveaux, l'ensemble des frontières entre classes étant encodées par le niveau 0.

Les travaux de la Section 7.1 ont été le sujet d'une publication en revue internationale :

— Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « Segment-before-Detect : Vehicle Detection and Classification through Semantic Segmentation of Aerial Images ». Dans : *Remote Sensing* 9.4 (13 avr. 2017), p. 368. doi : 10.3390/rs9040368

Les travaux de la Section 7.2 ont été le sujet d'une publication en conférence nationale :

— Nicolas Audebert et al. « Segmentation Sémantique Profonde Par Régression Sur Cartes de Distances Signées ». Dans : *Reconnaissance Des Formes, Image, Apprentissage et Perception (RFIAP)*. Marne-la-Vallée, France, juin 2018. URL : https://hal.archives-ouvertes.fr/hal-01809991 (visité le 27/08/2018)

Références

[1] Nicolas Audebert, Bertrand Le Saux et Sébastien Lefèvre. « Segment-before-Detect : Vehicle Detection and Classification through Semantic Segmentation of Aerial Images ». Dans : *Remote Sensing* 9.4 (13 avr. 2017), p. 368. DOI : 10.3390/rs9040368 (cf. p. 179).

[2] Nicolas Audebert et al. « Segmentation Sémantique Profonde Par Régression Sur Cartes de Distances Signées ». Dans : *Reconnaissance Des Formes, Image, Apprentissage et Perception (RFIAP)*. Marne-la-Vallée, France, juin 2018. URL : https://hal.archives-ouvertes.fr/hal-01809991 (cf. p. 180).

[3] Vijay Badrinarayanan, Alex Kendall et Roberto Cipolla. « SegNet : A Deep Convolutional Encoder-Decoder Architecture for Scene Segmentation ». Dans : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.12 (déc. 2017), p. 2481-2495. ISSN : 0162-8828. DOI : 10.1109/TPAMI.2016.2644615 (cf. p. 174, 178).

[4] Min Bai et Raquel Urtasun. « Deep Watershed Transform for Instance Segmentation ». Dans : *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Juil. 2017, p. 2858-2866. DOI : 10.1109/CVPR.2017.305 (cf. p. 166).

[5] Gedas Bertasius, Jianbo Shi et Lorenzo Torresani. « Semantic Segmentation With Boundary Neural Fields ». Dans : *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016, p. 3602-3610. URL : https://www.cv-foundation.org/openaccess/content_cvpr_2016/html/Bertasius_Semantic_Segmentation_With_CVPR_2016_paper.html (cf. p. 171).

[6] Serge Beucher et Fernand Meyer. « The Morphological Approach to Segmentation : The Watershed Transformation. Mathematical Morphology in Image Processing. » Dans : *Optical Engineering* 34 (1993), p. 433-481 (cf. p. 166).

[7] Gabriel J. Brostow, Julien Fauqueur et Roberto Cipolla. « Semantic Object Classes in Video : A High-Definition Ground Truth Database ». Dans : *Pattern Recognition Letters*. Video-based Object and Event Analysis 30.2 (15 jan. 2009), p. 88-97. ISSN : 0167-8655. DOI : 10.1016/j.patrec.2008.04.005. URL : http://www.sciencedirect.com/science/article/pii/S0167865508001220 (cf. p. 174).

[8] Manuel Campos-Taberner et al. « Processing of Extremely High-Resolution LiDAR and RGB Data : Outcome of the 2015 IEEE GRSS Data Fusion Contest Part A : 2-D Contest ». Dans : *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 9.12 (déc. 2016), p. 5547-5559. ISSN : 1939-1404. DOI : 10.1109/JSTARS.2016.2569162 (cf. p. 175, 177, 178).

[9] Liang-Chieh Chen et al. « DeepLab : Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs ». Dans : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40.4 (avr. 2018), p. 834-848. ISSN : 0162-8828. DOI : 10.1109/TPAMI.2017.2699184 (cf. p. 171, 178).

[10] Xueyun Chen et al. « Vehicle Detection in Satellite Images by Hybrid Deep Convolutional Neural Networks ». Dans : *IEEE Geoscience and Remote Sensing Letters* 11.10 (oct. 2014), p. 1797-1801. ISSN : 1545-598X. DOI : 10.1109/LGRS.2014.2309695 (cf. p. 160).

[11] Marius Cordts et al. « The Cityscapes Dataset for Semantic Urban Scene Understanding ». Dans : *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, United States, juin 2016, p. 3213-3223. DOI : 10.1109/CVPR.2016.350 (cf. p. 174).

[12] Nicolas Courty et al. « Optimal Transport for Domain Adaptation ». Dans : *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2016). URL : https://hal.archives-ouvertes.fr/hal-01377220 (cf. p. 170).

[13] Jifeng Dai, Kaiming He et Jian Sun. « Instance-Aware Semantic Segmentation via Multi-Task Network Cascades ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, United States, 2016, p. 3150-3158. DOI : 10.1109/CVPR.2016.343 (cf. p. 166, 171).

[14] Line Eikvil, Lars Aurdal et Hans Koren. « Classification-Based Vehicle Detection in High-Resolution Satellite Images ». Dans : *ISPRS Journal of Photogrammetry and Remote Sensing* 64.1 (jan. 2009), p. 65-72. ISSN : 09242716. DOI : 10.1016/j.isprsjprs.2008.09.005. URL : http://linkinghub.elsevier.com/retrieve/pii/S092427160800097X (cf. p. 160).

[15] Mark Everingham et al. « The Pascal Visual Object Classes Challenge : A Retrospective ». Dans : *International Journal of Computer Vision* 111.1 (25 juin 2014), p. 98-136. ISSN : 0920-5691, 1573-1405. DOI : 10.1007/s11263-014-0733-5. URL : http://link.springer.com/article/10.1007/s11263-014-0733-5 (cf. p. 164, 174).

[16] Joshua Gleason et al. « Vehicle Detection from Aerial Imagery ». Dans : *Robotics and Automation (ICRA), 2011 IEEE International Conference On*. IEEE, 2011, p. 2065-2070. URL : http://ieeexplore.ieee.org/abstract/document/5979853/ (cf. p. 160).

[17] Zeeshan Hayder, Xuming He et Mathieu Salzmann. « Boundary-Aware Instance Segmentation ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017 (cf. p. 171).

[18] Caner Hazirbas et al. « FuseNet : Incorporating Depth into Semantic Segmentation via Fusion-Based CNN Architecture ». Dans : *Computer Vision – ACCV 2016*. Asian Conference on Computer Vision. Springer, Cham, 20 nov. 2016, p. 213-228. DOI : 10.1007/978-3-319-54181-5_14 (cf. p. 175, 177).

[19] Kaiming He et al. « Delving Deep into Rectifiers : Surpassing Human-Level Performance on ImageNet Classification ». Dans : *Proceedings of the IEEE International Conference on Computer Vision*. IEEE International Conference on Computer Vision (ICCV). Déc. 2015, p. 1026-1034. DOI : 10.1109/ICCV.2015.123 (cf. p. 175).

[20] Kaiming He et al. « Deep Residual Learning for Image Recognition ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, United States, juin 2016, p. 770-778. DOI : 10.1109/CVPR.2016.90 (cf. p. 168, 174, 175).

[21] Kaiming He et al. « Mask R-CNN ». Dans : *Proceedings of the International Conference on Computer Vision*. International Conference on Computer Vision (ICCV). 20 mar. 2017 (cf. p. 166, 171).

[22] Ashley C. Holt et al. « Object-Based Detection and Classification of Vehicles from High-Resolution Aerial Photography ». Dans : *Photogrammetric Engineering & Remote Sensing* 75.7 (2009), p. 871-880. URL : http://www.ingentaconnect.com/content/asprs/pers/2009/00000075/00000007/art00007 (cf. p. 160).

[23] Bohao Huang et al. « Large-Scale Semantic Classification : Outcome of the First Year of Inria Aerial Image Labeling Benchmark ». Dans : *2018 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. 22 juil. 2018. URL : https://hal.inria.fr/hal-01767807/document (cf. p. 176, 177).

[24] Pranam Janney et David Booth. « Pose-Invariant Vehicle Identification in Aerial Electro-Optical Imagery ». Dans : *Machine Vision and Applications* 26.5 (1er juil. 2015), p. 575-591. ISSN : 0932-8092, 1432-1769. DOI : 10.1007/s00138-015-0687-9. URL : https://link.springer.com/article/10.1007/s00138-015-0687-9 (cf. p. 160).

[25] Simon Jégou et al. « The One Hundred Layers Tiramisu : Fully Convolutional DenseNets for Semantic Segmentation ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Honolulu, United States, juil. 2017, p. 1175-1183. DOI : 10.1109/CVPRW.2017.156 (cf. p. 175, 177, 178).

[26] Eric Jones, Travis Oliphant, Pearu Peterson et al. *SciPy : Open Source Scientific Tools for Python*. 2001-. URL : http://www.scipy.org/ (cf. p. 175).

[27] Dmitri Kamenetsky et Jamie Sherrah. « Aerial Car Detection and Urban Understanding ». Dans : *2015 International Conference on Digital Image Computing : Techniques and Applications (DICTA)*. 2015 International Conference on Digital Image Computing : Techniques and Applications (DICTA). Nov. 2015, p. 1-8. DOI : 10.1109/DICTA.2015.7371225 (cf. p. 160, 166).

[28] Alexander Kirillov et al. « Panoptic Segmentation ». Dans : (2 jan. 2018). arXiv : 1801.00868 [cs]. URL : http://arxiv.org/abs/1801.00868 (cf. p. 179).

[29] Alex Krizhevsky, Ilya Sutskever et Geoffrey E. Hinton. « ImageNet Classification with Deep Convolutional Neural Networks ». Dans : *Proceedings of the Neural Information Processing Systems (NIPS)*. NIPS. 2012, p. 1097-1105. URL : http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf (cf. p. 161, 167).

[30] TT. Hoang Ngan Le et al. « Reformulating Level Sets as Deep Recurrent Neural Network Approach to Semantic Segmentation ». Dans : *IEEE Transactions on Image Processing* 27.5 (mai 2018), p. 2393-2407. ISSN : 1057-7149. DOI : 10.1109/TIP.2018.2794205 (cf. p. 171).

[31] Franz Leberl et al. « Recognizing Cars in Aerial Imagery to Improve Orthophotos ». Dans : *GIS : Proceedings of the ACM International Symposium on Advances in Geographic Information Systems*. 1er jan. 2007, p. 2. DOI : 10.1145/1341012.1341015 (cf. p. 160).

[32] Yann LeCun et al. « Gradient-Based Learning Applied to Document Recognition ». Dans : *Proceedings of the IEEE* 86.11 (nov. 1998), p. 2278-2324. ISSN : 0018-9219. DOI : 10.1109/5.726791 (cf. p. 161, 167).

[33] Ziwei Liu et al. « Deep Learning Markov Random Field for Semantic Segmentation ». Dans : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40.8 (août 2018), p. 1814-1828. ISSN : 0162-8828. DOI : 10.1109/TPAMI.2017.2737535 (cf. p. 171).

[34] Emmanuel Maggiori et al. « Can Semantic Labeling Methods Generalize to Any City ? The Inria Aerial Image Labeling Benchmark ». Dans : *Proceedings of the IEEE International Symposium on Geoscience and Remote Sensing (IGARSS)*. IEEE International Symposium on Geoscience and Remote Sensing (IGARSS). 23 juil. 2017. DOI : 10.1109/IGARSS.2017.8127684. URL : https://hal.inria.fr/hal-01468452/document (cf. p. 174, 177).

[35] Dimitrios Marmanis et al. « Classification With an Edge : Improving Semantic Image Segmentation with Boundary Detection ». Dans : *ISPRS Journal of Photogrammetry and Remote Sensing* (2017). DOI : 10.1016/j.isprsjprs.2017.11.009. arXiv : 1612.01337 (cf. p. 160).

[36] Calvin R. Maurer, Rensheng Qi et Vijay Raghavan. « A Linear Time Algorithm for Computing Exact Euclidean Distance Transforms of Binary Images in Arbitrary Dimensions ». Dans : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 25.2 (fév. 2003), p. 265-270. ISSN : 0162-8828. DOI : 10.1109/TPAMI.2003.1177156 (cf. p. 173).

[37] Julien Michel et al. « Local Feature Based Supervised Object Detection : Sampling, Learning and Detection Strategies ». Dans : *2011 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, juil. 2011, p. 2381-2384. ISBN : 978-1-4577-1003-2. DOI : 10.1109/IGARSS.2011.6049689. URL : http://ieeexplore.ieee.org/document/6049689/ (cf. p. 160, 166).

[38] Keiller Nogueira, Otávio Penatti et Jefersson A. dos Santos. « Towards Better Exploiting Convolutional Neural Networks for Remote Sensing Scene Classification ». Dans : (3 fév. 2016). arXiv : 1602.01517 [cs]. URL : http://arxiv.org/abs/1602.01517 (cf. p. 161, 167).

[39] Jean Ogier du Terrail et Frédéric Jurie. « On the Use of Deep Neural Networks for the Detection of Small Vehicles in Ortho-Images ». Dans : *Proceedings of the International Conference on Image Processing (ICIP)*. 2017 IEEE International Conference on Image Processing (ICIP). Sept. 2017, p. 4212-4216. DOI : 10.1109/ICIP.2017.8297076 (cf. p. 160).

[40] Otávio Penatti, Keiller Nogueira et Jefersson A. dos Santos. « Do Deep Features Generalize from Everyday Objects to Remote Sensing and Aerial Scenes Domains ? » Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). Juin 2015, p. 44-51. DOI : 10.1109/CVPRW.2015.7301382 (cf. p. 167).

[41] *PyTorch : Tensors and Dynamic Neural Networks in Python with Strong GPU Acceleration*. http ://pytorch.org/. 2016-. URL : http://pytorch.org/ (cf. p. 175).

[42] Xiaojuan Qi et al. « 3D Graph Neural Networks for RGBD Semantic Segmentation ». Dans : *Proceedings of the International Conference on Computer Vision*. International Conference on Computer Vision (ICCV). 2017. URL : http://openaccess.thecvf.com/content_iccv_2017/html/Qi_3D_Graph_Neural_ICCV_2017_paper.html (cf. p. 176, 177).

[43] Hicham Randrianarivo, Bertrand Le Saux et Marin Ferecatu. « Urban Structure Detection with Deformable Part-Based Models ». Dans : *2013 IEEE International Geoscience and Remote Sensing Symposium - IGARSS*. 2013 IEEE International Geoscience and Remote Sensing Symposium - IGARSS. Juil. 2013, p. 200-203. DOI : 10.1109/IGARSS.2013.6721126 (cf. p. 160, 165).

[44] Hicham Randrianarivo et al. « Contextual Discriminatively Trained Model Mixture for Object Detection in Aerial Images ». Dans : *International Conference on Big Data from Space (BiDS'16)*. Spain, mar. 2016 (cf. p. 160, 164, 166).

[45] Sébastien Razakarivony et Frédéric Jurie. « Vehicle Detection in Aerial Imagery : A Small Target Detection Benchmark ». Dans : *Journal of Visual Communication and Image Representation* 34 (2016), p. 187-203. DOI : 10.1016/j.jvcir.2015.11.002. URL : http://www.sciencedirect.com/science/article/pii/S1047320315002187 (cf. p. 160, 162).

[46] Joseph Redmon et al. « You Only Look Once : Unified, Real-Time Object Detection ». Dans : *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Juin 2016, p. 779-788. doi : 10.1109/CVPR.2016.91 (cf. p. 160, 165).

[47] Shaoqing Ren et al. « Faster R-CNN : Towards Real-Time Object Detection with Region Proposal Networks ». Dans : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.6 (juin 2017), p. 1137-1149. issn : 0162-8828. doi : 10.1109/TPAMI.2016.2577031 (cf. p. 160, 165).

[48] Franz Rottensteiner et al. « The ISPRS Benchmark on Urban Object Classification and 3D Building Reconstruction ». Dans : *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 1 (2012), p. 3. url : https://t3sec3.rrzn.uni-hannover.de/cmsv021a.rrzn.uni-hannover.de/uploads/tx_tkpublikationen/isprsannals-I-3-293-2012.pdf (cf. p. 174).

[49] Olga Russakovsky et al. « ImageNet Large Scale Visual Recognition Challenge ». Dans : *International Journal of Computer Vision* 115.3 (11 avr. 2015), p. 211-252. issn : 0920-5691, 1573-1405. doi : 10.1007/s11263-015-0816-y. url : http://link.springer.com/article/10.1007/s11263-015-0816-y (cf. p. 161).

[50] Jamie Sherrah. « Fully Convolutional Networks for Dense Semantic Labelling of High-Resolution Aerial Imagery ». Dans : (8 juin 2016). arXiv : 1606.02585 [cs]. url : http://arxiv.org/abs/1606.02585 (cf. p. 164).

[51] Karen Simonyan et Andrew Zisserman. « Very Deep Convolutional Networks for Large-Scale Image Recognition ». Dans : *Proceedings of the International Conference on Learning Representations (ICLR)*. Mai 2015. url : http://arxiv.org/abs/1409.1556 (cf. p. 161, 167, 175).

[52] Lars Wilko Sommer, Tobias Schuchert et Jürgen Beyerer. « Fast Deep Vehicle Detection in Aerial Images ». Dans : *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*. 2017 IEEE Winter Conference on Applications of Computer Vision (WACV). Mar. 2017, p. 311-319. doi : 10.1109/WACV.2017.41 (cf. p. 160).

[53] Shuran Song, Samuel P. Lichtenberg et Jianxiong Xiao. « SUN RGB-D : A RGB-D Scene Understanding Benchmark Suite ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Juin 2015, p. 567-576. doi : 10.1109/CVPR.2015.7298655 (cf. p. 174).

[54] Nitish Srivastava et al. « Dropout : A Simple Way to Prevent Neural Networks from Overfitting ». Dans : *Journal of Machine Learning Research* 15 (2014), p. 1929-1958. url : http://jmlr.org/papers/v15/srivastava14a.html (cf. p. 168).

[55] Q. Tan, J. Wang et D. A. Aldred. « Road Vehicle Detection and Classification from Very-High-Resolution Color Digital Orthoimagery Based on Object-Oriented Method ». Dans : *IGARSS 2008 - 2008 IEEE International Geoscience and Remote Sensing Symposium*. IGARSS 2008 - 2008 IEEE International Geoscience and Remote Sensing Symposium. T. 4. Juil. 2008, p. IV - 475-IV - 478. doi : 10.1109/IGARSS.2008.4779761 (cf. p. 160).

[56] Tianyu Tang et al. « Vehicle Detection in Aerial Images Based on Region Convolutional Neural Networks and Hard Negative Example Mining ». Dans : *Sensors (Basel, Switzerland)* 17.2 (10 fév. 2017). issn : 1424-8220. doi : 10.3390/s17020336. pmid : 28208587. url : https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5335960/ (cf. p. 165).

[57] Devis Tuia, Claudio Persello et Lorenzo Bruzzone. « Domain Adaptation for the Classification of Remote Sensing Data : An Overview of Recent Advances ». Dans : *IEEE Geoscience and Remote Sensing Magazine* 4.2 (juin 2016), p. 41-57. ISSN : 2168-6831. DOI : 10.1109/MGRS.2016.2548504. URL : http://ieeexplore.ieee.org/document/7486184/ (cf. p. 170).

[58] Jonas Uhrig et al. « Pixel-Level Encoding and Depth Layering for Instance-Level Semantic Labeling ». Dans : *Pattern Recognition*. German Conference on Pattern Recognition. Lecture Notes in Computer Science. Springer, Cham, 12 sept. 2016, p. 14-25. ISBN : 978-3-319-45885-4 978-3-319-45886-1. DOI : 10.1007/978-3-319-45886-1_2. URL : https://link.springer.com/chapter/10.1007/978-3-319-45886-1_2 (cf. p. 171).

[59] Adam Van Etten. « You Only Look Twice : Rapid Multi-Scale Object Detection In Satellite Imagery ». Dans : (24 mai 2018). arXiv : 1805.09512 [cs]. URL : http://arxiv.org/abs/1805.09512 (cf. p. 160, 165).

[60] Q. Z. Ye. « The Signed Euclidean Distance Transform and Its Applications ». Dans : *Proceedings of the 9th International Conference on Pattern Recognition*. [1988 Proceedings] 9th International Conference on Pattern Recognition. 14-17 nov. 1988, 495-499 vol.1. DOI : 10.1109/ICPR.1988.28276 (cf. p. 172).

[61] Francisco de Assis Zampirolli et Leonardo Filipe. « A Fast CUDA-Based Implementation for the Euclidean Distance Transform ». Dans : *International Conference on High Performance Computing Simulation*. International Conference on High Performance Computing Simulation (HPCS). Juil. 2017, p. 815-818. DOI : 10.1109/HPCS.2017.123 (cf. p. 175).

[62] Hengshuang Zhao et al. « Pyramid Scene Parsing Network ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, United States, juil. 2017, p. 2881-2890. DOI : 10.1109/CVPR.2017.660. URL : http://openaccess.thecvf.com/content_cvpr_2017/html/Zhao_Pyramid_Scene_Parsing_CVPR_2017_paper.html (cf. p. 174, 175).

[63] Shuai Zheng et al. « Conditional Random Fields as Recurrent Neural Networks ». Dans : *Proceedings of the IEEE International Conference on Computer Vision*. IEEE International Conference on Computer Vision (ICCV). Déc. 2015, p. 1529-1537. DOI : 10.1109/ICCV.2015.179 (cf. p. 171).

[64] Weixun Zhou, Zhenfeng Shao et Qimin Cheng. « Deep Feature Representations for High-Resolution Remote Sensing Scene Classification ». Dans : *2016 4th International Workshop on Earth Observation and Remote Sensing Applications (EORSA)*. 2016 4th International Workshop on Earth Observation and Remote Sensing Applications (EORSA). Juil. 2016, p. 338-342. DOI : 10.1109/EORSA.2016.7552825 (cf. p. 161).

Chapitre 8

Conclusion et perspectives

It is good to have an end to journey toward ; but it is the journey that matters, in the end.

— Ursula Le Guin (The Left Hand of Darkness, 1969)

L'abondance nouvelle de données de télédétection est un véritable trésor pour la communauté scientifique. Grâce aux efforts décuplés investis dans les programmes d'observation de la Terre, nous disposons à présent d'images haute résolution sur l'ensemble du globe à des taux de revisite inégalés par le passé. En France, l'ensemble du territoire est imagé en aérien à 20 cm/px tous les 3 ans par l'IGN, tandis que le CNES produit une observation à 1,5 m/px de l'ensemble du pays chaque année grâce à SPOT. La constellation Sentinel-2 du programme européen Copernicus complète ce tableau par des acquisitions hebdomadaires à 10 m/px sur la Terre entière. À ces images viennent s'ajouter les données commerciales et de défense (WorldView, Pléiades), mais aussi les acquisitions radar, Lidar et hyperspectrales.

Les ressources humaines sont toutefois largement insuffisantes pour transformer cette masse de données brutes en information concrète. La photointerprétation manuelle est lente et coûteuse. Son automatisation représente un enjeu majeur pour les défis scientifiques d'aujourd'hui et de demain en écologie, en urbanisme, en météorologie ou encore en agriculture. Notre objectif dans cette thèse était de proposer des méthodes d'apprentissage statistique adaptées au problème de cartographie automatique. En nous appuyant sur les réseaux de neurones artificiels profonds, nous avons construit des modèles pour la sémantisation d'images aériennes et satellitaires à très haute et basse résolution pour une large gamme de capteurs optiques. Afin de tirer parti des données ancillaires comme les modèles numériques de terrain et les données géographiques ouvertes, nous avons en outre introduit des architectures multimodales de fusion de capteurs capables d'exploiter de l'information hétérogène riche. Enfin, nous avons montré les limites des approches sur des jeux de données limités ou massifs et proposé des alternatives, notamment par le biais de la génération de données synthétiques et l'introduction de *MiniFrance*, un nouveau jeu de données annoté à l'échelle de la France.

Ainsi, les travaux présentés dans ce manuscrit ont permis d'établir la place des réseaux profonds comme outil de choix pour l'interprétation automatique d'images de télédétection. Les performances de ces modèles sont désormais proches de ce qui est raisonnablement attendu par les experts thématiciens. En effet, la production automatique de cartes sémantiques à partir d'images optiques, qu'elles soient en couleur, multispectrales ou hyperspectrales, semble désormais réalisable à une échelle industrielle d'ici quelques années, y compris dans un cadre multimodal. Alors que l'apprentissage profond pour la télédétection était encore balbutiant il y a quelques années, il est désormais établi qu'il s'agit d'une approche incontournable. Cette thèse contribue à confirmer cette place mais surtout à étendre son champ d'application aux capteurs optiques inhabituels pour la communauté de la vision par ordinateur. Nous avons introduit des principes conducteurs pouvant servir de guide de bonnes pratiques pour l'application de réseaux convolutifs au vaste domaine de l'interprétation d'images d'observation de la Terre en étudiant les apports du préentraînement, des modules multiéchelles, de la fusion de données et de la régularisation des segmentations. Ces méthodologies encore peu explorées au début des années 2010 sont maintenant établies sur des fondements expérimentaux solides.

Dans un premier temps, nous avons montré que les méthodes de l'état de l'art utilisant la classification par région pouvaient être avantageusement remplacées par des approches

basées sur les réseaux de neurones entièrement convolutifs. En particulier, nous avons mis en évidence le rôle limitant des présegmentations non-supervisées pour l'apprentissage statistique et nous avons adapté les réseaux de segmentation sémantique aux images aériennes de télédétection IRRV et RVB. Nous avons par la suite étendu cette approche aux données satellitaires multispectrales Sentinel-2 en montrant expérimentalement la pertinence d'inclure les bandes hors du domaine visible afin de détecter plus de classes. Nous avons également étudié les approches d'apprentissage profond pour la classification de données hyperspectrales, en mettant notamment en avant l'efficacité des approches convolutives 3D sur les hypercubes. Ces contributions ont permis de produire des cartes à des niveaux de précision remplissant les besoins opérationnels pour les utilisateurs et consommateurs de données géographiques.

Dans un second temps, nous avons cherché à introduire de l'information auxiliaire dans les modèles orientés image afin de prendre en compte l'ensemble des données disponibles sur les scènes d'intérêt. En particulier, nous avons introduit deux architectures multimodales de réseaux convolutifs pour la segmentation sémantique permettant de fusionner des données issues de sources hétérogènes. En combinant d'une part images optiques et modèles numériques de terrain et, d'autre part, images optiques et données *OpenStreetMap*, nous avons pu diminuer le nombre d'erreurs commises par les réseaux profonds mis en œuvre pour la cartographie sémantique, notamment par le biais de la correction résiduelle. La prise en compte des données OSM, entièrement nouvelle, encourage en outre à étudier les possibilités de mise à jour automatique de ce type de base de données par des processus de *bootstrapping*.

Par la suite, nous nous sommes intéressés au comportement des modèles statistiques sur des jeux de données à petite et grande échelles. Dans le cas de l'imagerie hyperspectrale, nous avions en effet constaté que peu de données annotées étaient disponibles pour l'apprentissage de réseaux profonds. Nous avons donc conçu des modèles génératifs utilisant le principe des GAN pour la synthèse de spectres artificiels, avec succès. En outre, nous avons validé nos modèles de cartographie sémantique sur des jeux de données à grande échelle présentant des profils variés, notamment en constituant notre propre jeu de données haute résolution couvrant de nombreuses agglomérations du territoire français. Nous avons notamment montré que les réseaux que nous avons considéré étaient en mesure de passer à l'échelle et généralisaient à des scènes très diversifiées. L'introduction du jeu de données *MiniFrance* permettra à terme le préentraînement de modèles supervisés à une échelle encore jamais atteinte pour l'interprétation d'images de télédétection.

Enfin, nous avons étudié les techniques permettant de structurer les cartes sémantiques produites par les réseaux, en particulier dans le cadre de l'analyse d'image orientée objet. Nous avons conçu une méthode dite de segmentation avant détection permettant de réaliser une localisation et une reconnaissance fine des types de véhicules présents dans des images aériennes. Notre approche se fondant sur la segmentation permet notamment d'obtenir les formes des véhicules en plus de leur localisation et génère moins de fausses alarmes que les méthodes de l'état de l'art en télédétection. En outre, nous avons proposé une formulation alternative du problème général de segmentation sémantique sous forme de régression de cartes de distance, afin de structurer implicitement les cartes produites par les réseaux profonds. Cette approche nous a permis d'améliorer la qualité visuelle et statistique des cartes en contraignant le modèle à apprendre les relations spatiales entre objets de la scène. En particulier, la spatialisation des prédictions pixelliques permettra à terme de réaliser de façon unifiée une segmentation panoptique des images, aussi bien multimédia que de télédétection, permettant d'identifier conjointement les instances d'objets mais aussi les surfaces non-structurées. Cela ouvre la voie à de nouvelles pistes de recherche peu ou pas abordées jusqu'ici. Si cette thèse s'est concentrée sur la sémantisation des images, il est indispensable de replacer l'analyse géographique dans son contexte temporel. Du point de vue applicatif, c'est en effet l'évolution des cartes qui est intéressante, aussi bien pour la surveillance des typhons que le suivi de la déforestation. Qu'il s'agisse de détecter les changements entre deux acquisitions ou produire un suivi régulier d'une scène, la génération

systématique de cartes complètes pour chaque acquisition est coûteuse. À l'inverse, des méthodes de comparaison d'images ou d'études des séries temporelles, notamment par le biais de réseaux récurrents, permettraient de réaliser une cartographie incrémentale prenant en compte l'historique d'une scène, de façon plus économe mais aussi plus expressive.

Également, il est souhaitable de poursuivre les efforts de traitement de données hétérogènes afin de faire évoluer les architectures multimodales vers le traitement conjoint d'images et de données parcimonieuses ou non structurées, comme des nuages de points, des images au sol ou des annotations textuelles. Les premiers efforts existent en ce sens au travers d'architectures de réseaux modulaires pouvant exploiter de multiples sources de données, robustes à la perte d'une ou plusieurs modalités. Un des défauts couramment soulignés des capteurs optiques étant leur sensibilité aux conditions météorologiques, la prise en compte des données SAR serait notamment une avancée importante pour la cartographie automatisée à haute fréquence. La nature complexe des signaux radar, éloignée du processus habituel de capture d'image, nécessite toutefois des approches spécifiques capables de rendre justice à la physique de tels capteurs. En interprétation de scènes, la prise en compte de la géométrie, soit par reconstruction depuis l'image, soit par intégration d'un capteur type Lidar, permet d'envisager la génération de modèles 3D sémantiques. En télédétection, cela encourage ainsi à se pencher sur des approches combinant géométrie et sémantique pour la photogrammétrie et notamment la génération de modèles de surface et l'orthorectification.

Enfin, si l'apprentissage profond permet d'obtenir d'excellents résultats en pratique, l'interprétabilité des résultats pour les utilisateurs finaux demeure un enjeu majeur. Les représentations apprises par ces modèles statistiques sont difficilement exploitables par l'humain et ne véhiculent qu'une information limitée. L'interprétabilité des modèles, afin de permettre aux spécialistes de comprendre les prédictions des modèles, est un prérequis à la collaboration entre les utilisateurs finaux et la machine. Cela nécessite d'une part d'associer les représentations construites par le réseau à des concepts sémantiques manipulables par l'humain, et d'autre part à rendre explicable le processus de décision en rendant transparent les facteurs ayant eu le plus d'influence. En particulier, cela faciliterait l'apprentissage actif, permettant d'inclure la connaissance de l'humain et de bénéficier aussi bien de son expertise thématique que de ses habitudes de travail.

Annexe

A Jeux de données

One accurate measurement is worth a thousand expert opinions.

— Grace Hopper

A.1 Jeux de données en télédétection

Il existe plusieurs jeux de données pour la classification d'images optiques de télédétection. Citons ainsi les jeux de données *UC Merced* [13] contenant 2 100 images aériennes dans 21 classes d'occupation des sols, *Brazilian Coffe* [8] d'images SPOT pour la classification de terrains cultivés et SAT-4/SAT-6 [1] contenant respectivement 500 000 et 405 000 images aériennes pour plusieurs classes d'occupation des sols. L'inconvénient de ces jeux de données est d'une part la faible taille des images (256×256 px pour *UC Merced* et *Brazilian Coffe*, 28×28 px pour SAT) et la faible quantité d'annotations. En effet, ces jeux de données, prévus pour la classification, ne peuvent que difficilement être utilisés pour la segmentation sémantique. Cependant, plusieurs jeux de données comprenant des annotations denses ont été proposés.

A.1.1 ISPRS 2D Semantic Labeling

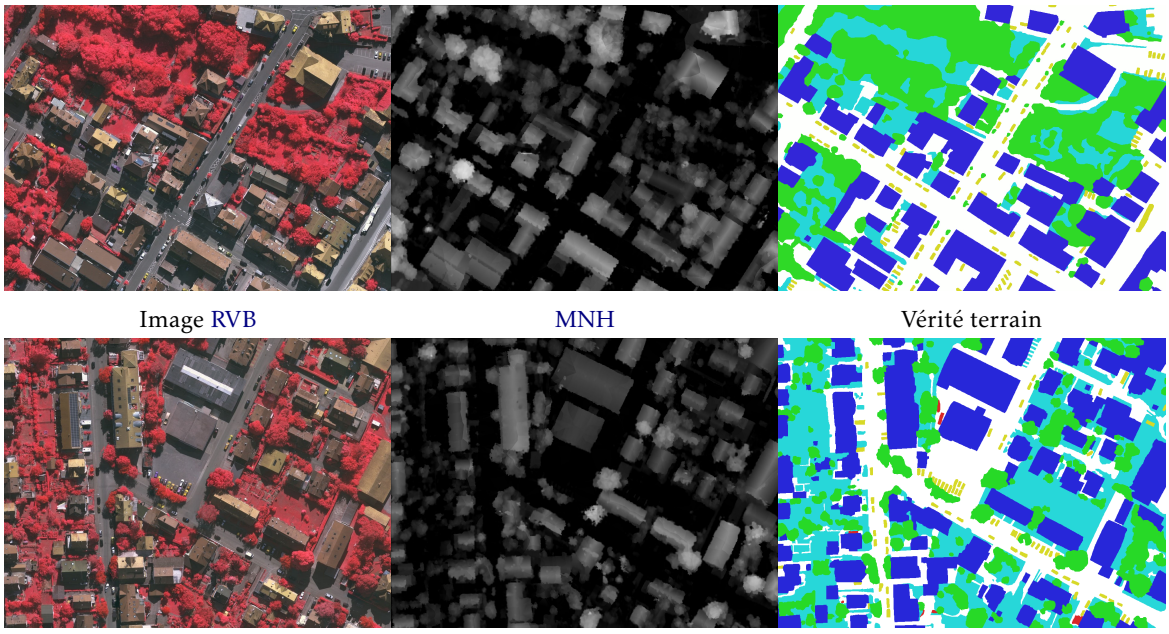


Figure A.1 – Images ortho-rectifiées et MNH pour le jeu de données ISPRS Vaihingen.

Le jeu de données ISPRS 2D Semantic Labeling [11] est constitué de deux ensembles d'images aériennes extrêmement haute résolution (EHR) fournies par le groupe de travail WG II/4 de l'*International Society for Photogrammetry and Remote Sensing*. Dans les deux cas, il s'agit de scènes urbaines disposant de cinq classes d'intérêt pour la segmentation

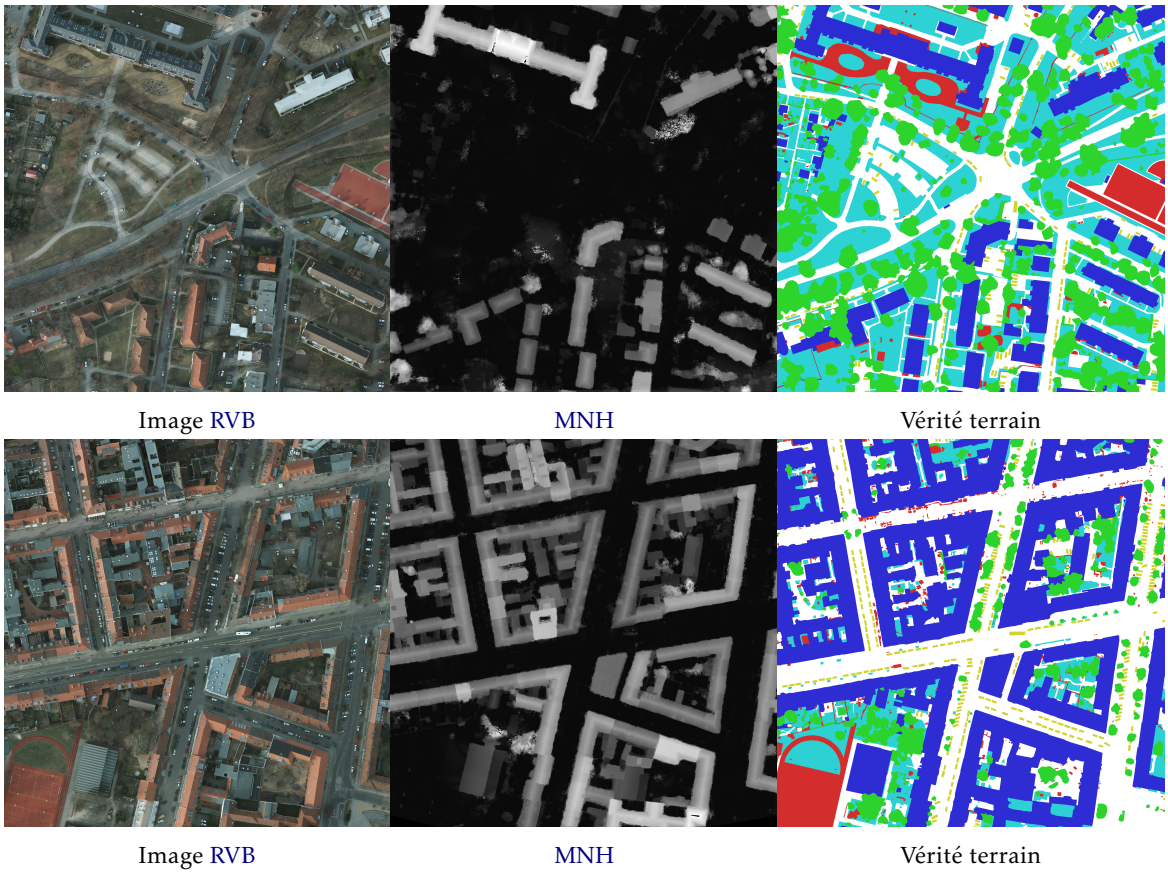


Figure A.2 – Images ortho-rectifiées et MNH pour le jeu de données ISPRS Potsdam.

sémantique : surfaces imperméables (routes, parkings, trottoirs...), bâtiments, végétation basse, arbres et véhicules. Une classe de rejet est également définie et comprend le mobilier urbain (bancs, poubelles, conteneurs...) et les surfaces inclassables (terrains de basketball, zones en travaux, points d'eau...).

Le jeu de données se décline en deux scènes. La première est issue d'une acquisition aéroportée sur la ville de Vaihingen (Allemagne) et comporte une mosaïque de 33 tuiles IRRV ortho-rectifiées à une résolution de 9 cm/px. L'acquisition optique est accompagnée d'une acquisition Lidar à la même résolution, dont a été extrait un MNE. Un MNH pré-calculé [4] dérivé du MNE est également disponible. Les ortho-images sont fournies en format TIFF à valeurs entières encodées sur 8 bits, tandis que le MNE est à valeurs réelles à virgules flottantes sur 32 bits. Toutes les données ont été recalées sur la même grille de pixels. Les images ont une taille moyenne d'environ 2600×1900px, soit une surface d'approximativement 40 000 m^2. Vaihingen est une ville de taille moyenne (28 853 habitants en 2009), caractérisée par une urbanisation moyenne composée majoritairement de pavillons résidentiels et d'espace verts urbains.

La seconde scène est une acquisition aéroportée sur la ville de Potsdam (Allemagne) et comporte une mosaïque de 38 tuiles IRRVB à une résolution de 5 cm/px réalisée par BSF Swissphoto. Le jeu de données est fourni par la *Deutsche Gesellschaft für Photogrammetrie, Fernerkundung und Geoinformation* (DGPF) [1]. Les tuiles présentent toutes les mêmes dimensions, à savoir 6000×6000px, soit une surface de 90 000 m^2. Un MNE et un MNH dérivé sont également fournis. Des annotations denses sont disponibles pour les mêmes classes que précédemment sur 24 images. À nouveau, l'ensemble des modalités sont recalées sur la même grille de pixels et les images sont fournies en TIFF à valeurs entières sur 8 bits, tandis que les modèles de surface sont fournis en valeurs réelles sur 32 bits. Potsdam est une ville

1. http://www.ifp.uni-stuttgart.de/dgpf/DKEP-Allg.html

urbanisée relativement grande (161 468 habitants en 2013), caractérisée par de nombreux immeubles, un réseau routier dense. À noter la présence d'un canal et de nombreux travaux de construction à la date de l'acquisition des images.

Quelques exemples représentatifs des deux acquisitions sont montrés dans les Figures A.1 et A.2. Le nombre de pixels dans chaque classe est détaillé dans la Figure A.10.

Les images dont les annotations ne sont pas rendues publiques servent à évaluer en aveugle les méthodes proposées par la communauté. La commission WG II/4 de l' ISPRS gère ainsi un tableau de résultat public [2] [3], détaillant les performances obtenues par différentes méthodes de l'état-de-l'art.

A.1.2 Data Fusion Contest 2015

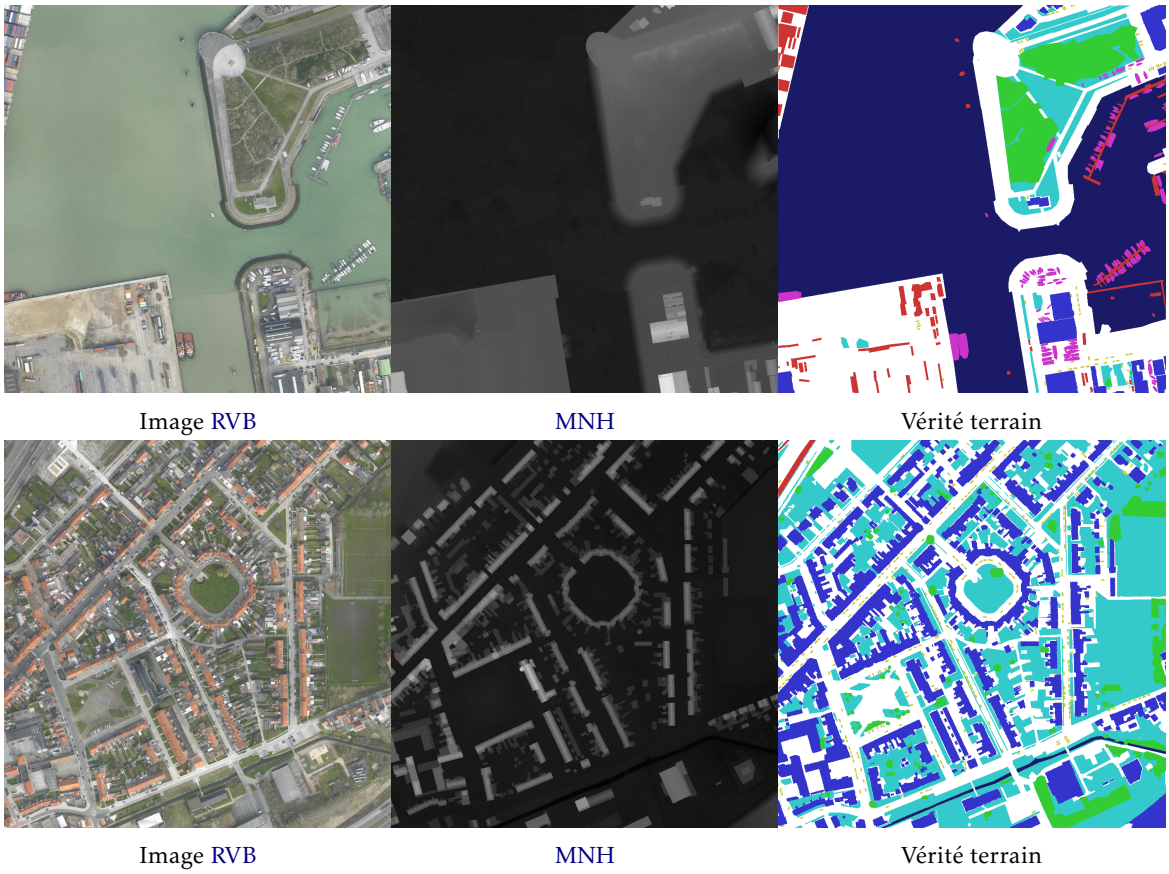


Image RVB | MNH | Vérité terrain

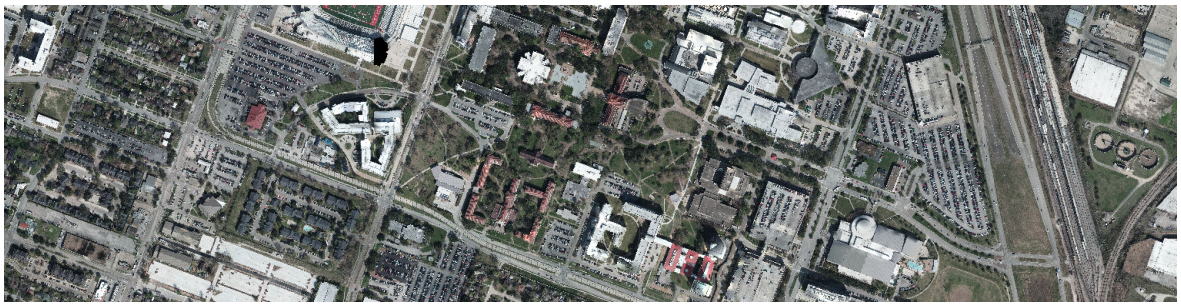
Image RVB | MNH | Vérité terrain

Figure A.3 – Images ortho-rectifiées et MNH pour le jeu de données DFC 2015.

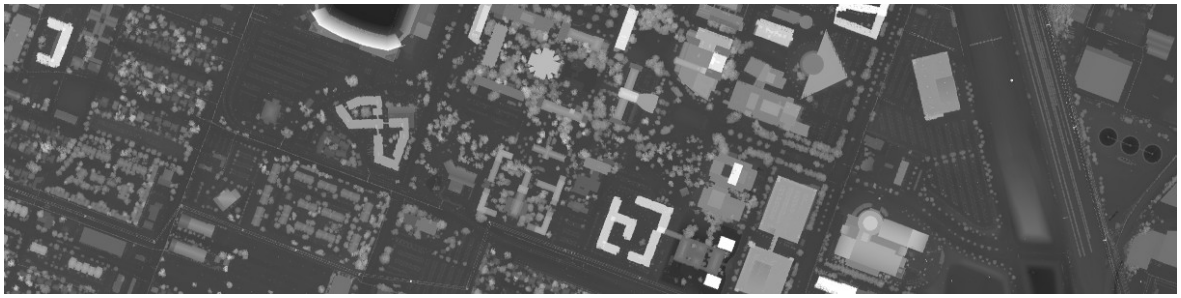
Le jeu de données DFC 2015 [3] est issu d'une compétition de fusion de données organisée par le groupe de travail GRSS de l'IEEE. Ce jeu de données comporte une mosaïque de 7 images couleurs ortho-rectifiées de dimensions 10000×10000px à une résolution au sol de 5 cm/px, soit une surface par tuile de 250 000 m^2. L'acquisition a été réalisée sur la zone portuaire de Zeebruges (Belgique) en mars 2011 par le département Communication, Information, Systèmes & Senseurs (CISS) de l'École royale militaire de Belgique. Il est accompagné d'une acquisition Lidar comprenant environ 65 points/m^2 espacés chacun de 10 cm. Les données couleurs sont fournies en TIFF (entiers 8 bits) et les données Lidar sont fournis rastérisées sous la forme d'un MNE (flottants sur 32 bits), ainsi qu'un nuage de points. Il s'agit d'une scène urbaine présentant principalement des installations portuaires.

Des annotations denses ont été réalisées par l'Office national d'études et de recherches aérospatiales (ONERA) [5] pour les classes bateau, voiture, végétation basse, arbre, bâtiment,

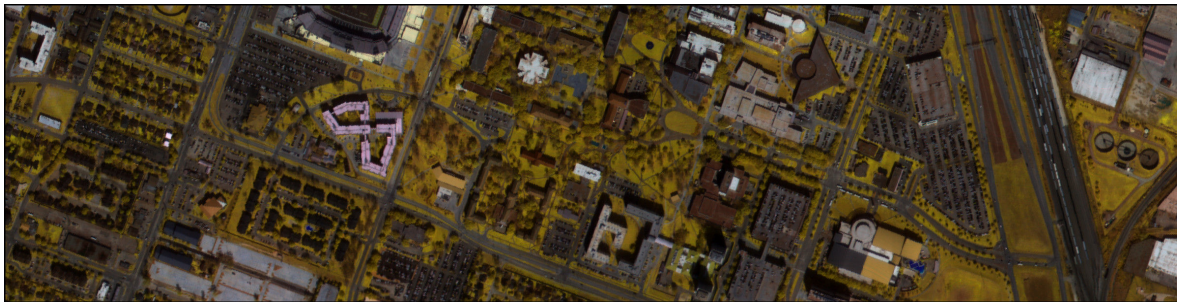
2. http://www2.isprs.org/commissions/comm2/wg4/vaihingen-2d-semantic-labeling-contest.html
3. http://www2.isprs.org/commissions/comm2/wg4/potsdam-2d-semantic-labeling.html



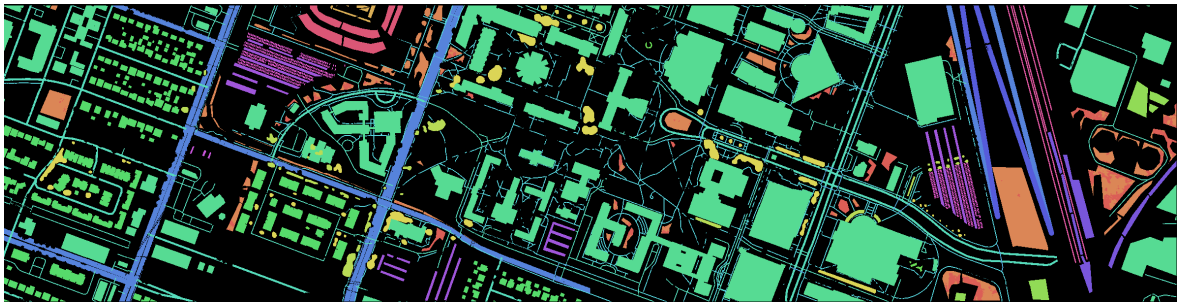
(a) Image RVB



(b) MNH



(c) Image hyperspectrale (fausses couleurs)



(d) Vérité terrain

Figure A.4 – Données d'entraînement du concours DFC 2018.

eau et surface imperméable. La Figure A.3 montre quelques exemples d'images extraites du jeu de données et la Figure A.11a détaille la répartition des pixels dans les différentes classes.

Un tableau de résultats public est maintenu par l'IEEE GRSS [4] afin de permettre la comparaison entre différentes méthodes de classification.

A.1.3 Data Fusion Contest 2018

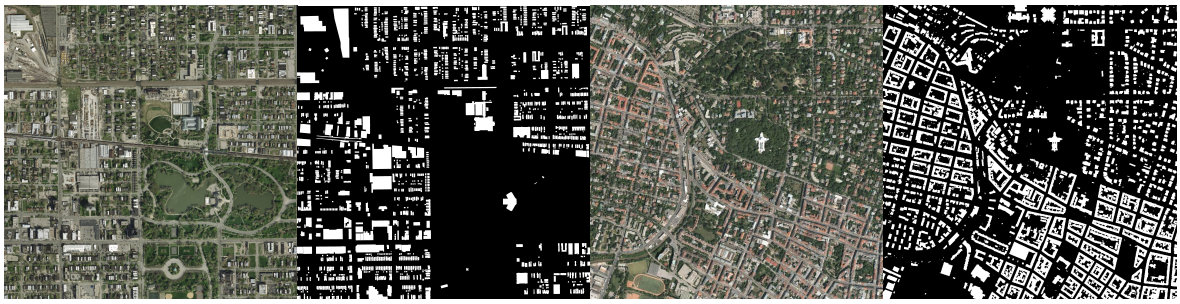
Le jeu de données DFC 2018 [6] est également issu de la compétition *Data Fusion Contest* (DFC) organisée par l'IEEE GRSS. Il comporte 14 images aériennes RVB ortho-rectifiées THR à 5 cm/px de dimensions 12000×12000px, accompagnées par une image hyperspectrale à 48

4. http://dase.ticinumaerospace.com/

bandes à 1 m/px entre 380 et 1050 nm. Une acquisition Lidar multispectrale est également disponible à une résolution de 0,5 m/px, dont est dérivé un MNH. L'ensembles des données sont géoréférencées et recalées. Les données ont été acquises par le *National Center for Airborne Laser Mapping* sur la ville de Houston (États-Unis) en février 2017. L'image concerne principalement l'université de Houston et ses alentours, par conséquent il s'agit d'une scène très urbanisée incluant des installations massives (gare et voies de chemins de fer, stade). Des annotations partielles sont fournies pour la moitié du jeu de données sur diverses classes d'intérêt urbaines, l'autre moitié des annotations étant conservées secrètes pour l'évaluation. Le jeu d'apprentissage est illustré par la Figure A.4 et la Figure A.11b détaille la répartition des pixels dans les différentes classes.

Un tableau de résultats public est maintenu par l'IEEE GRSS afin de permettre la comparaison entre différentes méthodes de classification.

A.1.4 Inria Aerial Image Labeling



Ortho-image (Chicago) Vérité terrain (Chicago) Ortho-image (Vienne) Vérité terrain (Vienne)

Figure A.5 – Exemples d'images extraites de la base de données *Inria Aerial Image Labeling*.

Le jeu de données *Inria Aerial Image Labeling* [7] contient 360 images RVB ortho-rectifiées de taille 5000 × 5000px à une résolution de 30 cm/px, soit une surface de 2,25 km^2. Les images ont été compilées depuis les bases de données de l'USGS pour Austin, Chicago, Kitsap County, Bellingham, Bloomington et San Francisco et depuis les services géographiques régionaux autrichiens pour le Tyrol, Vienne et Innsbruck. Il s'agit d'acquisitions aéroportées, ortho-rectifiées, ré-échantillonnées à 30 cm/px et fournies sous un format couleur 8 bits. Les annotations des empreintes de bâtiments sont obtenues à partir des sources cadastrales locales. La moitié des images peuvent être utilisées pour l'entraînement de modèles d'extraction de bâtiments, les annotations étant disponibles librement. Le reste du jeu de données est réservé à l'évaluation des modèles par les auteurs du jeu de données. Quelques images et annotations du jeu d'apprentissage sont illustrées dans la Figure A.5.

Parmi ces villes, plusieurs sont de grandes agglomérations caractérisées par une forte densité de bâtiments, mêlant habitations personnelles, constructions massives (gares, hôpitaux, usines…) et immeubles. À l'inverse, les autres zones présentent une densité de bâtiments modérée voire faible, avec un relief et une végétation importants pour le Tyrol. Cette diversité des profils de zones observées vise à permettre de mesurer la capacité des modèles à généraliser à différents environnements géographiques.

Les organisateurs gèrent un tableau de résultats public [5] permettant de comparer les performances obtenues par différentes méthodes.

A.1.5 VEDAI

La base de données *Vehicle Detection in Aerial Imagery* (VEDAI) [10] est une collection d'images aériennes ortho-rectifiées mises à disposition par l'*Automated Geographic Refe-*

5. https://project.inria.fr/aerialimagelabeling/leaderboard/



Figure A.6 – Extraits d'images annotées du jeu de données VEDAI.

rence Center de l'Utah. Les images, acquises au printemps 2012, ont une résolution au sol de 12,5 cm/px et comportent 4 canaux, RVB et infrarouge, encodés sur 8 bits. Les images originales ont été découpées en 1 210 tuiles de dimensions 1024 px × 1024 px. Une version sous-échantillonnée à 25 cm/px utilisant des tuiles de dimensions 512 px × 512 px est également disponible.

Les véhicules présents dans les images sont annotés à fin de détection par des boîtes englobantes ainsi qu'une étiquette correspondant au type de véhicule. Neuf classes d'intérêt sont ainsi identifiées : avion, bateau, camping-car, voiture, pick-up, tracteur, camion, van et une classe "autres". Les coordonnées du centre du véhicule et l'angle correspondant à son orientation principale sont également annotés.

Ce jeu de données couvre des zones principalement rurales avec une faible densité de véhicules. Les images présentent une large variété de contextes, comme des parkings, des aérodromes, des axes routiers plus ou moins importants, des champs et des habitations. Quelques exemples d'images et d'annotations correspondantes sont montrés dans la Figure A.6.

A.1.6 NZAM/ONERA Christchurch

Le jeu de données NZAM/ONERA Christchurch comporte 4 images RVB ortho-rectifiées à une résolution de 10 cm/px acquises après la séisme ayant frappé la ville de Christchurch en Nouvelle-Zélande le 22 février 2011. Les images sont distribuées sous licence Creative Commons Attribution 3.0 par le *New Zealand's Land Information Office* [6]. Chaque image, de dimensions ≈ 5000 × 4000px, a été annotée par l'ONERA/DTIS [9] pour les classes d'intérêt "bâtiments" (797 objets), "véhicules" (2357 objets) et "végétation" (938 objets). L'ensemble des objets sont annotés par une boîte englobante polygonale, ce qui en fait des annotations moins précises que les vérités terrain pixelliques des jeux de données ISPRS, par exemple. Des exemples sont donnés dans la Figure A.7.

6. http://www.linz.govt.nz/land/maps/linz-topographic-maps/imagery-orthophotos/christchurch-earthquake-imagery



Figure A.7 – Images et annotations extraites du jeu de données NZAM/ONERA Christchurch.

A.2 Jeux de données en interprétation de scènes

A.2.1 CamVid

La base de données CamVid (*Cambridge-driving Labeled Video Database*) [2] est une collection d'images extraite à 1 Hz d'une vidéo RVB de 10 minutes en situation de conduite automobile dans la ville de Cambridge (Royaume-Uni). 367 images d'entraînement et 233 images de test à une résolution de 360×480px ont été extraites des vidéos et manuellement annotées dans 11 classes d'intérêt telles que la route, les bâtiments, les autres véhicules, les piétons, les panneaux de signalisations, le trottoir, etc. Dans l'ensemble, il s'agit d'images embarquées de conduite à vitesse réduite dans un environnement urbain comprenant de nombreux objets mobiles. La Figure A.8 montre quelques exemples d'images annotées du jeu d'entraînement.

A.2.2 SUN RGB-D

Le jeu de données SUN RGB-D [12] comporte 10 335 images Red-Green-Blue + Depth d'intérieur acquises en utilisant plusieurs capteurs (Kinect, Xtion, RealSense). Chaque image est une paire couleur RVB et carte de profondeur et a été annotée au niveau pixel pour 37 classes d'intérêt d'objets ou de surfaces telles que "chaise", "sol", "mur" ou encore "table". Les images sont généralement redimensionnées à 224×224. En tout, 146 617 objets 2D sont annotés sous forme de polygones non recouvrants, permettant ainsi d'obtenir pour chaque scène une vérité terrain de segmentation sémantique. D'autres annotations sont également disponibles comme la catégorie de la scène 2,5D parmi les 47 disponibles ou 800 types d'objets 3D identifiés par une boîte englobante. Des exemples d'images et d'annotations sont



FIGURE A.8 – Images RVB (première ligne) et annotations pixelliques (deuxième ligne) extraites du jeu de données CamVid.



FIGURE A.9 – Images RVB, cartes de profondeur et annotations extraites du jeu de données SUN RGB-D.

données dans la Figure A.9.

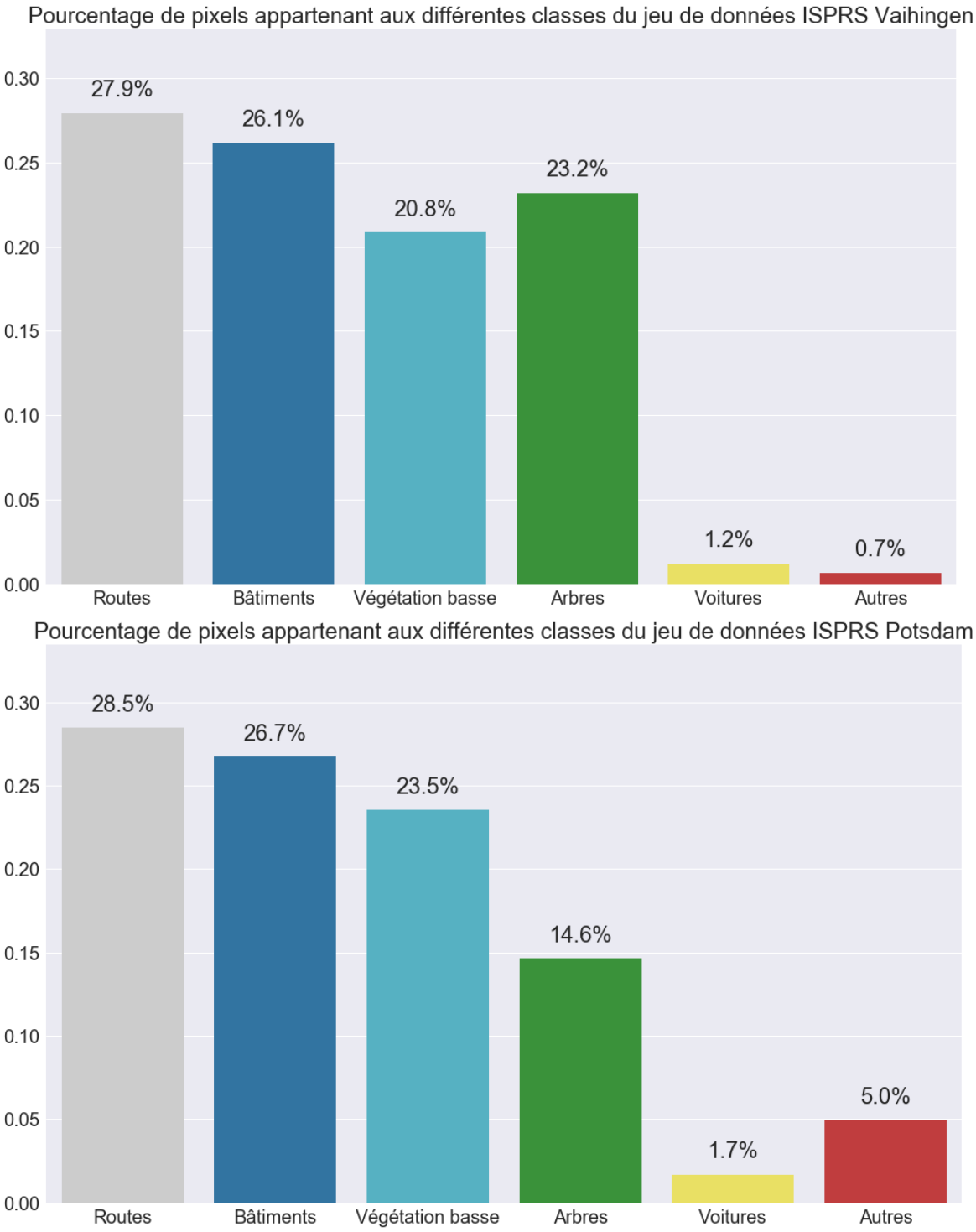
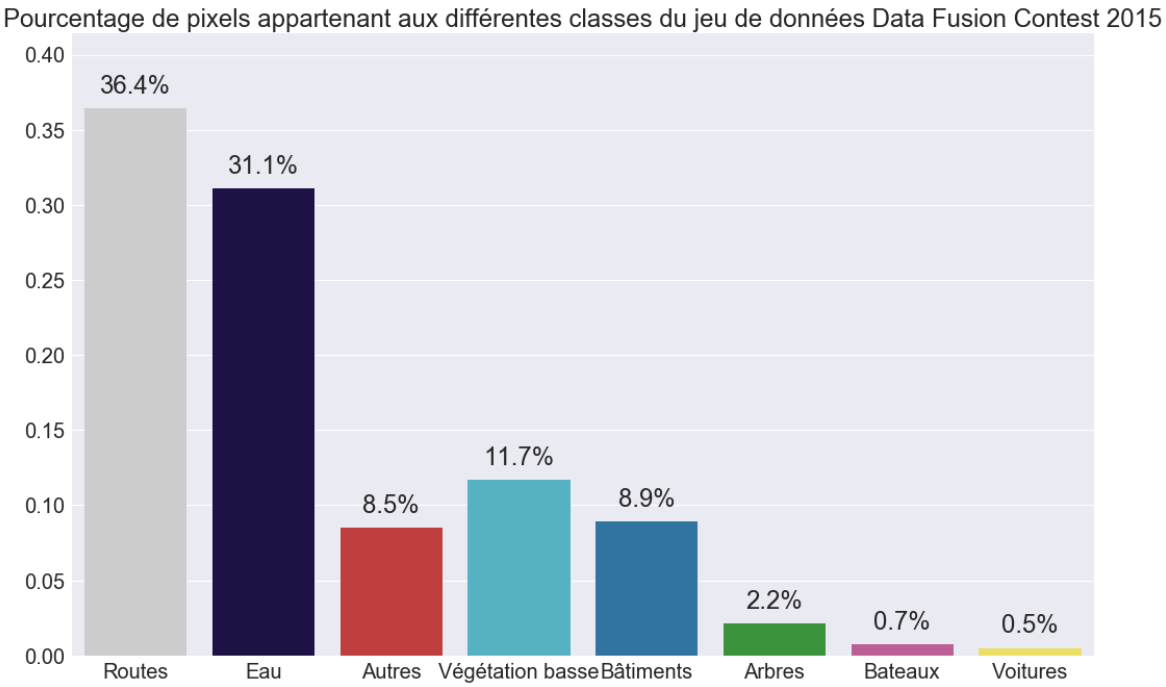
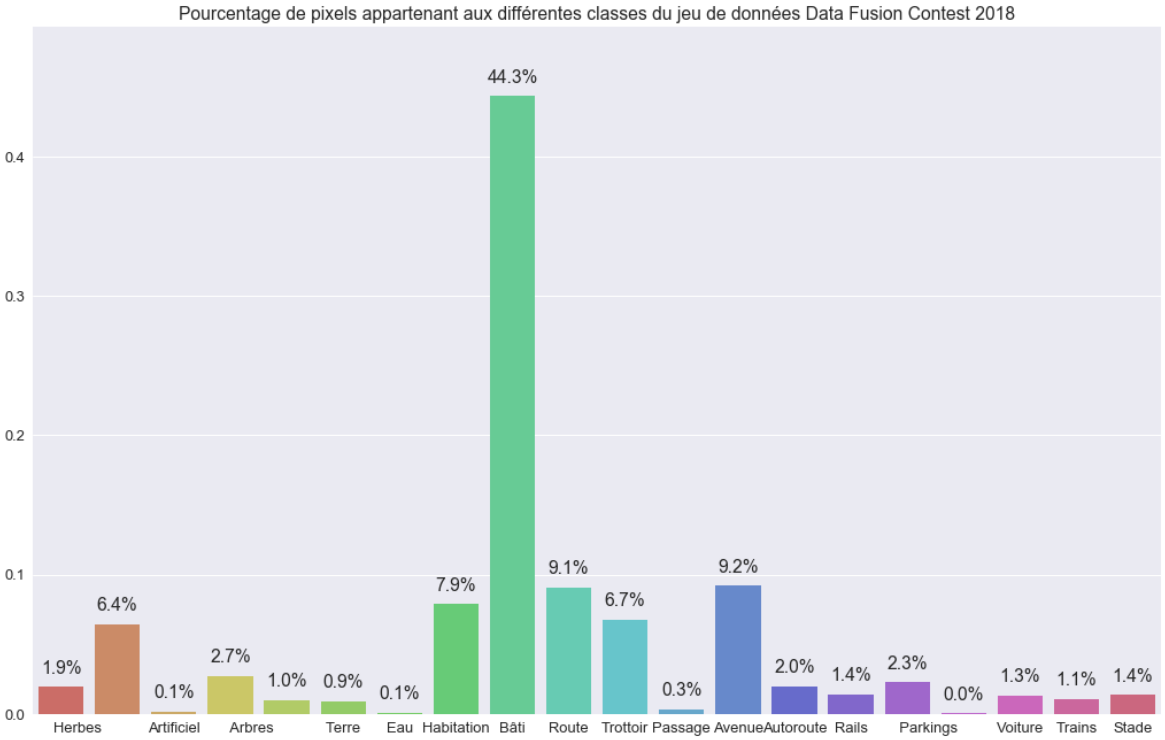


FIGURE A.10 – Répartition des pixels des jeux de données ISPRS dans différentes classes d'intérêt.



(a) Répartition des pixels du jeu de données DFC 2015.



(b) Répartition des pixels du jeu de données DFC 2018.

FIGURE A.11 – Répartition des pixels dans les jeux de données du *Data Fusion Contest*.

Bibliographie

[1] Saikat Basu et al. « DeepSat : A Learning Framework for Satellite Imagery ». Dans : *Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems*. SIGSPATIAL '15. New York, NY, USA : ACM, 2015, 37 :1-37 :10. ISBN : 978-1-4503-3967-4. DOI : 10.1145/2820783.2820816. URL : http://doi.acm.org/10.1145/2820783.2820816 (cf. p. I).

[2] Gabriel J. Brostow, Julien Fauqueur et Roberto Cipolla. « Semantic Object Classes in Video : A High-Definition Ground Truth Database ». Dans : *Pattern Recognition Letters*. Video-based Object and Event Analysis 30.2 (15 jan. 2009), p. 88-97. ISSN : 0167-8655. DOI : 10.1016/j.patrec.2008.04.005. URL : http://www.sciencedirect.com/science/article/pii/S0167865508001220 (cf. p. VII).

[3] Manuel Campos-Taberner et al. « Processing of Extremely High-Resolution LiDAR and RGB Data : Outcome of the 2015 IEEE GRSS Data Fusion Contest Part A : 2-D Contest ». Dans : *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 9.12 (déc. 2016), p. 5547-5559. ISSN : 1939-1404. DOI : 10.1109/JSTARS.2016.2569162 (cf. p. III).

[4] Markus Gerke. *Use of the Stair Vision Library within the ISPRS 2D Semantic Labeling Benchmark (Vaihingen)*. International Institute for Geo-Information Science and Earth Observation, 2015. URL : https://www.researchgate.net/profile/Markus_Gerke/publication/270104226_Use_of_the_Stair_Vision_Library_within_the_ISPRS_2D_Semantic_Labeling_Benchmark_(Vaihingen)/links/54ae59c50cf2828b29fcdf4b.pdf (cf. p. II).

[5] Adrien Lagrange et al. « Benchmarking Classification of Earth-Observation Data : From Learning Explicit Features to Convolutional Networks ». Dans : *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS). Juil. 2015, p. 4173-4176. DOI : 10.1109/IGARSS.2015.7326745 (cf. p. III).

[6] Bertrand Le Saux et al. « 2018 IEEE GRSS Data Fusion Contest : Multimodal Land Use Classification [Technical Committees] ». Dans : *IEEE Geoscience and Remote Sensing Magazine* 6.1 (mar. 2018), p. 52-54. ISSN : 2473-2397. DOI : 10.1109/MGRS.2018.2798161 (cf. p. IV).

[7] Emmanuel Maggiori et al. « Can Semantic Labeling Methods Generalize to Any City ? The Inria Aerial Image Labeling Benchmark ». Dans : *Proceedings of the IEEE International Symposium on Geoscience and Remote Sensing (IGARSS)*. IEEE International Symposium on Geoscience and Remote Sensing (IGARSS). 23 juil. 2017. DOI : 10.1109/IGARSS.2017.8127684. URL : https://hal.inria.fr/hal-01468452/document (cf. p. V).

[8] Otávio Penatti, Keiller Nogueira et Jefersson A. dos Santos. « Do Deep Features Generalize from Everyday Objects to Remote Sensing and Aerial Scenes Domains ? » Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). Juin 2015, p. 44-51. DOI : 10.1109/CVPRW.2015.7301382 (cf. p. I).

[9] Hicham Randrianarivo, Bertrand Le Saux et Marin Ferecatu. « Urban Structure Detection with Deformable Part-Based Models ». Dans : *2013 IEEE International Geoscience and Remote Sensing Symposium - IGARSS*. 2013 IEEE International Geoscience and Remote Sensing Symposium - IGARSS. Juil. 2013, p. 200-203. doi : 10.1109/IGARSS.2013.6721126 (cf. p. VI).

[10] Sébastien Razakarivony et Frédéric Jurie. « Vehicle Detection in Aerial Imagery : A Small Target Detection Benchmark ». Dans : *Journal of Visual Communication and Image Representation* 34 (2016), p. 187-203. doi : 10.1016/j.jvcir.2015.11.002. url : http://www.sciencedirect.com/science/article/pii/S1047320315002187 (cf. p. V).

[11] Franz Rottensteiner et al. « The ISPRS Benchmark on Urban Object Classification and 3D Building Reconstruction ». Dans : *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 1 (2012), p. 3. url : https://t3sec3.rrzn.uni-hannover.de/cmsv021a.rrzn.uni-hannover.de/uploads/tx_tkpublikationen/isprsannals-I-3-293-2012.pdf (cf. p. I).

[12] Shuran Song, Samuel P. Lichtenberg et Jianxiong Xiao. « SUN RGB-D : A RGB-D Scene Understanding Benchmark Suite ». Dans : *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Juin 2015, p. 567-576. doi : 10.1109/CVPR.2015.7298655 (cf. p. VII).

[13] Yi Yang et Shawn Newsam. « Bag-of-Visual-Words and Spatial Extensions for Land-Use Classification ». Dans : *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*. GIS '10. New York, NY, USA : ACM, 2010, p. 270-279. isbn : 978-1-4503-0428-3. doi : 10.1145/1869790.1869829. url : http://doi.acm.org/10.1145/1869790.1869829 (cf. p. I).

Annexe

B Code

It's still magic even if you know how it's done.

— Terry Pratchett (A Hat Full of Sky, 2004)

B.1 FCN pour la cartographie sémantique

Site Internet : https://github.com/nshaud/DeepNetsForEO

Ce code sous licence libre implémente le réseau SegNet de référence du Chapitre 3 pour la segmentation sémantique d'images RVB et multispectrales. Écrit en langage Python, ce logiciel utilise la bibliothèque Pytorch afin d'exécuter l'entraînement et l'inférence du modèle indépendamment sur GPU ou CPU. Les paramètres configurables permettent de reproduire les expériences décrites dans les Chapitres 3 à 5 sur des images couleur ou multispectrales.

B.2 *DeepHyperX*

Site Internet : https://gitlab.inria.fr/naudeber/DeepHyperX

Ce code sous licence libre est la boîte à outils modulaire pour la classification d'images hyperspectrales décrite dans le Chapitre 4. Écrit en Python, ce logiciel utilise les bibliothèques Pytorch et scikit-learn. Il s'adresse à deux types de public :

— Les concepteurs de modèles souhaitant implémenter et valider de nouvelles architectures neuronales dans un cadre expérimental standardisé,

— Les utilisateurs thématiques désirant utiliser des réseaux de neurones de l'état de l'art pour classifier leurs données.

B.3 *MiniFrance*

Site Internet : https://gitlab.inria.fr/naudeber/FranceDataset

Ce code sous licence libre rassemble les scripts déployés pour la constitution du jeu de données *MiniFrance* décrit dans le Chapitre 6. Écrit en Python et en bash, ces scripts permettent de convertir les images de la BD ORTHO et de rasteriser les données du cadastre français ainsi que de *UrbanAtlas* sur la même mosaïque grâce aux bibliothèques rasterio, fiona et geopandas.

B.4 HyperGANs

Site Internet : https://github.com/nshaud/HyperGANs

Ce code sous licence libre est une implémentation de référence du Wasserstein-GAN pour la génération de spectres synthétiques décrit dans le Chapitre 6. Écrit en Python, ce logiciel utilise la bibliothèque Pytorch. Il permet de reproduire les expériences en génération de spectres décrites dans ce manuscrit sur divers jeux de données.

Annexe C

Liste des acronymes

ACP analyse en composantes principales. v, 99, 101, 102, 141, 142

AVIRIS *Airborne Visible/Infrared Imaging Spectrometer*. 97, 98

BN normalisation par lot, ou *Batch Normalization*. 21, 27, 66

CDS carte de distance signée. 167–169, 171, 173–175

CIELAB L'espace L*a*b* CIE 1976, ou CIELAB, est un espace de représentation de couleurs utilisant la clarté L$*$, dérivée de la luminance, et $a*$ et $b*$ comme paramètres exprimant l'écart colorimétrique par rapport à une surface grise. L'intérêt de cette représentation est qu'une distance euclidienne dans l'espace CIELAB est proche de la distance colorimétrique perçue par l'œil humain. 59, 62

CNES Centre national d'études spatiales. XVII, 3, 181

CNN *Convolutional Neural Network*, réseau de neurones dont certaines couches effectuent des convolutions à poids partagés. v, viii, 12, 13, 23, 24, 28, 29, 31, 32, 35, 42, 57, 64, 73, 78, 94, 100–106, 116, 117, 119, 128, 138, 156–158, 162–165

CPU *Central Processing Unit*, processeur de calcul générique d'un ordinateur. XIII, XIX, 103, 171

CRF champ de Markov conditionnel, ou *Conditional Random Field*. 35, 36, 42, 43, 78, 100, 118, 122, 160

CUDA *Compute Unified Device Architecture*, technologie permettant de programmer sur GPU depuis le langage C. 13

DBN *Deep Belief Networks*. 12, 15, 101, 117

DFC *Data Fusion Contest*. vi–viii, III–V, X, 97, 103–105, 173

EHR Extrêmement haute résolution. Désigne une image de télédétection d'une résolution inférieure au sol à $10cm$. 42, 43, 58, 72, 94, 145

ELU *Exponential Linear Unit*. 15

ERS Algorithme de segmentation par marche aléatoire. 59

FCN *Fully Convolutional Network*, réseau de neurones entièrement convolutif conservant la structure spatiale grâce à l'absence de couches entièrement connectées. iv, 34–36, 43, 65, 66, 68, 72, 74, 78–80, 87, 88, 90, 92, 102, 105, 118, 122, 128, 130, 146, 155, 156, 160, 161, 166, 167, 175

FH Algorithme de segmentation d'image proposé par Felzenszwalb et Huttenlocher. 59, 60, 62, 63

GAN modèle génératif utilisant deux réseaux de neurones entraînés en concurrence. v, XIII, 138–145, 182

GMM modèle de mélange gaussien, ou *Gaussian Mixture Model*,. 138

GPU *Graphics Processing Unit*, processeur de calcul hautement parallèle originellement conçu pour le rendu graphique, puis réutilisé pour le calcul scientifique matriciel, notamment dans le cas des réseaux convolutifs profonds. XIII, XV, XIX, 12, 13, 35, 91, 103, 121, 159

GRSS *Geoscience & Remote Sensing Society*. III–V, 103, 105

HOG *Histograms of Oriented Gradients* (HOG) ou histogrammes de gradients orientés. Approximation discrète de la distribution locale du gradient dans une image selon une direction pré-définie. 12, 22, 34, 40, 64, 156, 161

HR Haute résolution. Désigne une image de télédétection d'une résolution inférieure au sol à $1m$. 42, 66

HSeg *Hierarchial Segmentation*. Algorithme de segmentation hiérarchique. 61, 62, 72, 73, 75

IEEE *Institute of Electrical and Electronics Engineers*. III–V, 103, 105

IGN Institut national de l'information géographique et forestière. 3, 147, 181

ILSVRC *ImageNet Large-Scale Visual Recognition Challenge*. XIX, 12, 13, 30–33, 163

IRRV Image trois canaux dans le proche infrarouge, le rouge et le vert. v, II, 58, 62, 72, 74, 76–80, 88, 89, 105, 107, 108, 119, 122, 123, 125, 126, 145, 146, 160, 171, 182

IRRVB Image multispectrale dans le proche infrarouge, le rouge, le vert et le bleu. II, 40, 87–89, 130, 156

ISPRS *International Society for Photogrammetry and Remote Sensing*. iv–viii, I–III, VI, IX, 63, 72–81, 88, 89, 106–108, 122, 123, 125–130, 137, 145, 146, 148, 158–160, 164, 170–175

IsU Intersection sur union. Mesure de performance d'un classifieur utilisée notamment dans le cadre de la segmentation sémantique. Elle correspond au rapport du nombre d'échantillons étant positifs à la fois pour la classifieur et en réalité et du nombre d'échantillons étant positifs dans le classifieur ou dans la réalité. 71, 159–162, 172–174

JPEG *Joint Photographic Experts Group*. 22

Lidar *Light Detection And Ranging*, technique de mesure de distance utilisant le temps parcouru par un faisceau lumineux entre son émission et la réception de son écho. II, III, V, 6, 37, 39, 40, 105–107, 115, 123, 181

LRN *Local Response Normalization*. 26, 30

LSC *Linear Spectral Clustering*. Algorithme de segmentation superpixels. 60, 72–75

MNE Modèle Numérique d'Élévation. Description de la topographie d'une surface terrestre prenant en compte les objets surélevés. Parfois également nommé modèle numérique de surface (MNS) dans la littérature. iv, vii, II, III, 39, 96, 105–108, 119

MNH Modèle Numérique de Hauteur. Mesure de la hauteur des points surélevés par normalisation par rapport au terrain sous-jacent. v–vii, I–V, 39, 79, 105–108, 122–124, 126

MNT Modèle Numérique de Terrain. Description de la topographie d'une surface terrestre, ne prenant pas en compte les objets surélevés. iv, 39, 105, 106, 118

MRS *Multi-Resolution Segmentation*. Algorithme de segmentation multi-échelles implémenté dans le logiciel eCognition. 60–63, 72, 73

NDVI *Normalized Difference Vegetation Index*. 40, 99, 105, 107, 108, 118

NDWI *Normalized Difference Water Index*. 40, 99

NN *Neural Network*. 104

ONERA Office national d'études et de recherches aérospatiales. IV

OSM *OpenStreetMap*. v, vii, 127–131, 182

PReLU *Parametrized Rectified Linear Unit*. 15

RBM *Restricted Boltzmann Machines*. 12, 101

ReLU *Rectified Linear Unit*. 12, 13, 15, 66, 103, 140

RF *Random Forest*, ou forêt aléatoire,. 41, 78

RGB-D Red-Green-Blue + Depth. VII, VIII, XIX, 77, 117–119

RNN *Recurrent Neural Network*. 101, 103, 104

RVB Espace de représentation des images naturelles sous forme de trois canaux rouge, vert et bleu. v, vi, I–VIII, XIII, 6, 29, 37, 38, 40, 58–60, 62, 80, 81, 87–89, 91, 92, 94, 96, 101, 102, 107, 117–119, 123, 126–128, 130, 131, 147, 157, 160, 161, 163, 164, 172, 173, 182

SAR radar à synthèse d'ouverture (en anglais *Synthetic Aperture Radar*). XIX, 37, 39, 40, 61, 181

SCALP *Superpixels with Contour Adherence using Linear Path*. Algorithme de segmentation de type superpixels. 60

SEEDS *Superpixels Extracted via Energy-Driven Sampling*. Algorithme de segmentation superpixels. 61

SIFT Les descripteurs SIFT *(Scale-Invariant Feature Transform)* sont des caractéristiques images calculées sur des points d'intérêt cherchant une invariance à l'échelle, l'angle de vue et à la luminosité. 12, 22, 34, 64

SIG système d'information géographique. 39, 127, 130, 162

SLIC *Simple Linear Iterative Clustering*. Algorithme de segmentation superpixels. 60–63, 72, 73

SPOT Satellites Pour l'Observation de la Terre, une famille de satellites de télédétection français conçus par le CNES. I, 2, 88, 147, 181

SVM *Support Vector Machine*, en français machine à vecteurs de support, parfois sous la dénomination Séparateur à Vaste Marge. Outil de classification ou de régression. vii, 12, 41, 58, 64, 65, 72, 73, 100–104, 118, 141, 142, 144, 156, 161

THR Très haute résolution. Désigne une image de télédétection d'une résolution inférieure au sol à 50*cm*. I, V, 42, 58, 156

TIFF *Tagged Image File Format*, format de fichier image supportant l'ajout d'informations de géoréférencement grâce au standard GeoTIFF. II, III

USGS *United States Geological Survey*. 138

VEDAI *Vehicle Detection in Aerial Imagery*. v, vi, viii, VI, 156, 158, 159, 162, 164–166

Annexe D

Glossaire

AlexNet Architecture de réseau de neurones convolutif de classification d'images particulièrement populaire, arrivée en tête de la compétition ILSVRC en 2012. 29, 64, 72, 117, 157, 162

Caffe Bibliothèque logicielle C++ dotée d'interfaces Python et Matlab pour l'apprentissage profond. 74

FuseNet Architecture de réseau de neurones entièrement convolutive multi-modale, dérivée de SegNet, pour la segmentation sémantique d'images RGB-D. 117, 119, 129

hyperspectral Imagerie utilisant des récepteurs sur plusieurs dizaines de longueurs d'onde, y compris hors du domaine visible. IV, XIII, 38, 39, 94, 181

ImageNet Base de données contenant plus d'un million d'images annotées représentant mille classes d'objet, comportant aussi bien des voitures que des chats, des chiens, des chaises, des personnes. . .. 12, 57, 146, 147

Landsat Premier programme spatial d'observation de la Terre, initié par la NASA en 1972. 2, 41, 88

multispectral Imagerie utilisant des récepteurs sur plusieurs longueurs d'onde. v, XIII, 37, 39, 61, 88, 91–94

Pléiades Paire de satellites optiques très haute résolution français. 38, 88

PyTorch Bibliothèque logicielle C++/Python de calcul tensoriel sur CPU et GPU, spécialisée pour l'apprentissage profond. 74, 91, 103

ResNet Architecture de réseau de neurones convolutif utilisant le paradigme d'apprentissage résiduel. 32, 66, 119

scikit-learn Bibliothèque logicielle Python d'apprentissage automatique. 103

SegNet Architecture de réseau de neurones entièrement convolutive pour la segmentation sémantique. XIX, 66, 72, 88, 106, 119, 145, 156, 171

Sentinel Programme spatial d'observation de la Terre européen, démarré en 2014 avec les satellites SAR Sentinel-1A/B. La famille Sentinel comporte également les satellites multispectraux Sentinel-2A/B lancés en 2015 et 2017, et les satellites Sentinel-3A/B lancés en 2016 et 2018 pour l'océanographie. 2, 87, 90, 181, 182

VGG-16 Architecture de réseau de neurones convolutif à 16 couches pour la classification d'images. 66, 128, 157, 162