



HAL
open science

Apprentissage semi-supervisé pour la compréhension des données d'observation de la Terre à large-échelle

Javiera Castillo-Navarro

► To cite this version:

Javiera Castillo-Navarro. Apprentissage semi-supervisé pour la compréhension des données d'observation de la Terre à large-échelle. Computer Vision and Pattern Recognition [cs.CV]. Université de Bretagne Sud, 2022. English. NNT : 2022LORIS622 . tel-03909116

HAL Id: tel-03909116

<https://theses.hal.science/tel-03909116>

Submitted on 21 Dec 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE DE DOCTORAT DE

L'UNIVERSITÉ DE BRETAGNE SUD

ÉCOLE DOCTORALE N° 601
*Mathématiques et Sciences et Technologies
de l'Information et de la Communication*
Spécialité : *Informatique*

Par

Javiera CASTILLO NAVARRO

Semi-supervised learning for large-scale Earth observation data understanding

Thèse présentée et soutenue à Paris, le 23 mars 2022
Préparée à l'Institut de recherche en informatique et systèmes aléatoires (UMR CNRS 6074)
et l'Office nationale d'études et de recherches aérospatiales.
Thèse N° : 622

Rapporteurs avant soutenance :

Nicolas THOME Professor – Conservatoire National des Arts et Métiers
Xiao Xiang ZHU Professor – Technical University of Munich

Composition du Jury :

Examineurs :	Marie CHABERT	Professor – Institut National Polytechnique de Toulouse
	Felipe TOBAR	Associate Professor – Universidad de Chile
	Devis TUIA	Associate Professor – École Polytechnique Fédérale de Lausanne
Dir. de thèse :	Sébastien LEFÈVRE	Professor – Université Bretagne Sud
Encadrants :	Bertrand LE SAUX	Research scientist – European Space Agency
	Alexandre BOULCH	Research scientist – Valeo.ai

Invité :

Stéphane MAY Research scientist – Centre National d'Études Spatiales

ACKNOWLEDGEMENTS

“The most important things are the hardest things to say. They are the things you get ashamed of because words diminish your feelings - words shrink things that seem timeless when they are in your head to no more than living size when they are brought out.”

— Stephen King

There is truly no way I could express with words the gratitude I feel towards all of you who have made it possible that I arrive at this moment. Some of you will not be mentioned explicitly here because my memory is fragile and I have limited time and space to write these acknowledgments. However, I am sincerely grateful to everyone who has encouraged me to continue this path, and that with little gestures –a smile or small talk– have made me who I have become.

Je voudrais commencer mes remerciements avec quelques mots dédiés à mes encadrants : Bertrand Le Saux, Alexandre Boulch et Sébastien Lefèvre. Bertrand, merci beaucoup pour ta rigueur académique et pour ton perfectionnisme, tu m’as souvent montré que je suis capable de faire mieux. J’ai également apprécié nos conversations “hors science” (notamment pendant ma première année) qui ont donné un aspect plus humain à la thèse. Alexandre, merci pour avoir été toujours à la recherche de nouvelles idées et pour me montrer qu’il faut penser *out-of-the-box*. Merci également pour avoir été toujours disponible lorsque des problèmes pratiques arrivaient, tes conseils ont été précieux. Sébastien, toujours très réactif, à l’écoute et attentif, tu m’as apporté du calme dans les moments de stress. Merci aussi pour ta rigueur scientifique et pour me pousser à suivre les directions de recherche qui m’intéressent. Enfin, je pourrais continuer à vous remercier pour plein de choses. En résumé, merci –tous les trois– pour m’avoir accordé votre confiance pour mener ce projet de thèse et pour m’avoir accompagnée pendant ces trois ans, malgré les difficultés et la distance. Je vous suis très reconnaissante pour vos encouragements, vos conseils et votre bienveillance.

I would like to thank all the members of my thesis committee. In the first place, I express my gratitude toward Nicolas Thome and Xiaoxiang Zhu, reviewers of this work, who took the time to conscientiously read my thesis and made constructive comments

that certainly helped to improve this work. Secondly, I thank Marie Chabert, Felipe Tobar, and Devis Tuia for agreeing to participate in this event and for taking the time to evaluate my work.

I extend my acknowledgments to Germain Forestier, who –together with Nicolas Thome– was part of my *Comité de suivi de thèse* and took the time to discuss my work in our yearly meetings.

Je voudrais remercier ensuite toute l'équipe IVA. Parmi les permanents, je voudrais adresser quelques mots à : Philippe, c'est toujours un plaisir de discuter de montagne et randonnée, mais surtout merci pour ton intérêt sur les sujets d'actualité concernant le Chili ; Martial, merci pour ton aide avec les démarches administratives, grâce à toi je peux soutenir ma thèse ; Elise, merci pour ta bienveillance, pour prendre des nouvelles de temps en temps ; Adrien et Stéphane, pour me donner l'opportunité d'intervenir comme responsable de TP dans des cours de machine learning.

Je remercie les doctorants(tes) et jeunes docteurs(res): Pol, Adrien, Kevin, Marius, Quentin, Nathan (el chachador), Thomas, William, Rémy, Philip, Simon, Benjamin, Antoine, Louis, Alexis, Soufiane, Rodrigo, Pierre G. et Anthelme (quasiment un doctorant de plus). Nous avons partagé des idées, des discussions de la vie courante, des moments difficiles de la thèse, mais aussi des soirées et des jeux de société. Merci aussi pour tous les cafés, les viennoiseries, les gâteaux, les parties de tarot et de tamalou. Je garderai toujours à l'esprit la bonne ambiance, les blagues et les rires en salle de pause ; ils restent parmi mes meilleurs souvenirs de ces trois années. Guillaume (grand), merci pour ton aide à différents moments et pour les chocolats kinder et les petits gâteaux d'encouragement à la fin de la rédaction. Rodolphe et Guillaume (petit), merci pour vos mots et votre soutien à des moments particulièrement difficiles. Et Gaston, merci pour toutes nos discussions –parfois de science, parfois sur la thèse, parfois d'escalade et de montagne– et surtout, merci de m'avoir écouté.

Je remercie mes co-bureaux successifs : Marcela, qui m'a toujours aidée et soutenue pendant ma première année de thèse ; Laurane, tu es partie trop vite ! mais je garde des superbes souvenirs ; et finalement Maxime, parce que j'aurais difficilement survécu à la période d'octobre-décembre 2021 sans les schoko-bons et paillettes cachés dans le calendrier de l'avent. Je remercie aussi Camille, qui a été là, même avant que la thèse ne commence.

Merci au département de mathématiques de l'université de Versailles Saint-Quentin pour m'avoir permis d'intervenir en tant que chargée des TDs dans différents cours. Je

remercie également les étudiants (même s'ils et elles ne liront sûrement pas ces mots) pour rendre chaque cours une expérience unique.

Je salue avec amitié Nicolas Audebert, co-auteur de certains articles et qui a rendu la section sur MiniFrance possible¹.

Je remercie également l'équipe OBELIX à Vannes. Je regrette de ne pas avoir eu plus d'opportunités d'échanger avec vous, que ce soit d'un point de vue scientifique ou autre (je n'oublierai pas le char à voile).

Je remercie l'ONERA et le CNES qui ont apporté le financement pour que cette thèse voie le jour.

A los amigos de siempre, los que a pesar de no estar físicamente, están de alguna manera presentes y serán siempre importantes para mí: Cami y Lucero, qué hermoso que los años pasen y el cariño y confianza sigan intactos, gracias por nuestras profundas conversaciones este verano; Ove, mi overmana! gracias por escucharme, hacerme sentir acompañada y llamarme en los momentos más complejos; Pato, Emi, Pesce, Rubén, Romi, porque pese a estar repartidos por todo el mundo, sé que puedo contar con ustedes. Jaime, muchas gracias por nuestras conversaciones, por el concierto de 31 minutos! por las canciones y mucho más².

Aux amis grimpeurs, vous ne savez pas à quel point vous êtes importants pour moi, ni comment vous m'avez apporté pour que cette thèse puisse enfin être soutenue. Pendant le confinement, les seules sorties autorisées étaient celles dans la merveilleuse forêt de Fontainebleau, où nous avons partagé des journées ensoleillées, la pluie et même la neige. Merci Seb, Alex, Matt, Vincent, Davide (il primo porcino!), Laure, Corentin et Gaston (encore). Effectivement, l'escalade m'a permis de rester sereine pendant les périodes les plus stressantes de la thèse.

A los chilenos en Paris, que logran curar en 5 minutos hasta mis más grandes episodios de homesickness. Gracias Bere (técnicamente pas chilienne, mais chilienne pour nous), Riffo, Mile, Checo y Javi (aunque ya no estén). Garrido, ya son tantos años compartiendo y la verdad es que no tengo palabras para expresar mi gratitud, simplemente gracias por estar aquí. JP (actualmente no en París), gracias por todo el cariño y las confidencias, tu compañía fue *muy* importante en los últimos meses de redacción de este manuscrito.

A mis papis, Gran Pa y Súper Mami, que a mis ojos serán siempre mis súper héroes.

1. Oups ! je crois avoir lu une phrase similaire dans une thèse en 2018. Plus sérieusement, mes remerciements sont sincères.

2. Incluyendo la cita de Stephen King :).

A pesar de la distancia, siempre logran entregarme su cariño y consejos, como si viviera en el segundo piso de la casa. Gracias por nuestras llamadas de los domingos, que fueron esenciales para sobrevivir sin verlos durante dos años. Las palabras simplemente no son suficientes para agradecerles, si los abrazos se pudieran escribir, escribiría un texto infinito (y ni siquiera eso sería suficiente).

Tito, siempre serás mi modelo a seguir. Por diversas razones, siempre te voy a admirar. Gracias por permitirme a veces jugar el papel de hermana mayor, gracias por tu confianza, por tus consejos y por reírte de mis chistes fomes. Gracias también a Isa y Robertín por inspirarme siempre.

Je remercie à Martine et Yvon pour leur soutien, leurs conseils et toutes leurs attentions. Merci pour m'avoir accueillie avec tant d'affection dans votre famille.

Finalmente, merci Benoit, mi cariñito, pour être là. Durant ces trois années, tu as été bien plus que mon pololo, tu as été un ami, un co-bureau (merci covid (?)) et un relecteur de plusieurs textes ! Nous nous sommes soutenus mutuellement dans les différents moments de la vie d'un doctorant. D'ailleurs, cette thèse a été largement enrichie par nos discussions (parfois philosophiques) sur des papiers et sur différentes méthodes ! Gracias, simplemente, por tu compañía durante estos tres años, por el Lapino, y por todas las aventuras que nos quedan por vivir.

TABLE OF CONTENTS

Résumé en français	11
Introduction	21
Context	22
Objectives	25
Organization of this document	27
Publications	28
1 Related work	31
Chapter summary	32
1.1 Semantic segmentation	32
1.2 Learning paradigms	36
1.2.1 Supervised learning	37
1.2.2 Unsupervised learning	38
1.2.3 Semi-supervised learning	38
1.2.4 Discriminative vs. generative models	40
1.3 Deep learning	41
1.3.1 Brief history of deep neural networks	41
1.3.2 The multilayer perceptron	43
1.3.3 Convolutional neural networks	46
1.3.4 Fully convolutional networks	48
1.3.5 Deep semi-supervised learning	50
1.3.6 Deep learning in Earth observation	52
2 The potential of semi-supervised learning in Earth observation	57
Chapter summary	58
2.1 Current Earth observation benchmarks	59
2.2 The necessity of new training paradigms and large-scale EO datasets	62
2.2.1 Analysis of supervised learning on small-scale datasets	63

TABLE OF CONTENTS

2.2.2	Supervised learning at large-scale	65
2.3	The MiniFrance suite	70
2.3.1	MiniFrance	70
2.3.2	TinyMiniFrance	75
2.4	Statistical analysis of the representativeness of training and test datasets	76
2.4.1	Appearance analysis	77
2.4.2	Class representativeness analysis.	81
2.5	Defining the labeled, unlabeled and test splits for MiniFrance	85
2.6	Comparing MiniFrance to classic datasets	87
2.7	Data fusion contest 2022: MF-DFC22	88
2.8	Conclusions	92
3	Semi-supervised learning: discriminative approaches	95
	Chapter summary	96
3.1	Introduction: discriminative models	97
3.2	Semi-supervised learning cast as multi-task	97
3.2.1	Multi-task learning	98
3.2.2	Multi-task semantic segmentation networks	100
3.2.3	Auxiliary tasks and losses	102
3.2.4	Experiments	106
3.3	Semi-supervised learning through consistency regularization	119
3.3.1	Vicinal risk minimization	121
3.3.2	FixMatch	123
3.3.3	Experiments	127
3.4	Conclusions	129
4	Semi-supervised learning: generative approaches	131
	Chapter summary	132
4.1	Introduction: generative models	133
4.2	Energy-based models	138
4.2.1	Joint energy-based models (JEM)	140
4.2.2	Semi-supervised learning with JEM	142
4.3	Experiments	143
4.3.1	Joint classification and generation with JEM	145
4.3.2	Semi-supervised classification with JEM	147

4.3.3	Out-of-distribution analysis	150
4.3.4	Application to land cover mapping	152
4.3.5	Can we combine FixMatch and JEM?	155
4.4	Limitations	156
4.5	Perspectives: semantic segmentation with JEM	157
4.6	Conclusions	159
Conclusion		161
	Summary of contributions	161
	Perspectives for future work	164
Bibliography		167
List of Figures		190
List of Tables		193

Introduction

L'analyse des données d'observation de la Terre joue un rôle majeur dans la façon dont nous comprenons notre planète et son fonctionnement. En effet, la quantité toujours croissante de données d'imagerie de télédétection au cours des dernières décennies a permis de nouveaux développements dans les domaines de l'écologie, de l'urbanisme ou de la réponse aux catastrophes naturelles, et sera certainement cruciale dans la lutte contre le changement climatique, en surveillant en permanence la déforestation, l'élévation du niveau des mers et les émissions de gaz à effet de serre dans l'atmosphère.

Les technologies de télédétection nous permettent de voir ce que nous ne sommes pas en mesure d'observer de nos propres yeux. Elles permettent de recueillir des informations sur notre planète en exploitant le fait que les matériaux présents dans une scène reflètent, absorbent et émettent des rayonnements électromagnétiques de manière différente selon leur composition moléculaire et leur forme [1].

L'histoire des satellites d'observation de la Terre a commencé en 1947 avec Spoutnik 1. Depuis, les efforts se sont multipliés et, à ce jour, plus de 150 satellites d'observation de la Terre sont actuellement en orbite. Ces constellations de satellites ont fourni un déluge de données d'observation de la Terre, avec des volumes de stockage dépassant des dizaines de pétaoctets, acquérant plus de centaines de téraoctets de données par jour. Les satellites Sentinel-2 fournissent à eux seuls plus de 20 To de données par jour [2, 3]. L'interprétation et la compréhension de l'imagerie satellitaire nécessitent une certaine expertise du domaine, combinant la connaissance de la physique des capteurs et celle de l'application considérée. Cependant, l'exploitation de ces quantités gargantuesques de données n'est pas humainement possible. L'analyse automatique des images d'observation de la Terre semble être le seul moyen d'extraire les informations contenues dans ces données.

Au cours de la dernière décennie, les techniques d'apprentissage profond - et la croissance conséquente de la puissance de calcul - ont transformé les domaines de la

vision par ordinateur et du traitement des images. Plus récemment, l'apprentissage profond a démontré son potentiel pour résoudre les problèmes des sciences de la Terre et du climat [2, 4, 5]. Ces techniques représentent des outils prometteurs pour construire de nouveaux modèles orientés aux données pour l'observation de la Terre.

Malheureusement, la plupart des algorithmes d'apprentissage développés à ce jour dépendent fortement de la disponibilité de bases de données massives d'images annotées. En général, les données étiquetées sont difficiles à obtenir, ce qui nécessite des ressources, du temps et des connaissances spécialisées. De plus, il n'existe pas de moyen efficace de fournir des étiquettes annotées par des humains pour l'immensité des données de télédétection disponibles. D'autre part, les données brutes -sans étiquettes- sont abondantes, surtout en télédétection où les satellites génèrent des données en continu. Pour cette raison, nous sommes convaincus que les méthodes semi-supervisées -qui exploitent les données non étiquetées pour aider le processus d'apprentissage- seront essentielles pour pousser plus loin les capacités de généralisation des modèles.

Par conséquent, l'objectif principal de cette thèse est d'avancer sur la voie de la cartographie automatique à grande échelle. A cette fin, nous travaillons sur l'analyse et le développement de nouvelles méthodes semi-supervisées qui permettraient d'exploiter l'abondance d'images de télédétection non étiquetées et de surpasser l'état de l'art actuel basé sur des modèles entièrement supervisés.

Ce travail se situe alors à l'intersection de trois domaines : la télédétection pour l'observation de la Terre, l'apprentissage automatique et la vision par ordinateur. Cette thèse vise à développer des techniques d'apprentissage automatique (en particulier, l'apprentissage profond) et de vision par ordinateur au service des applications de télédétection, en utilisant des images d'observation de la Terre.

Objectifs

L'objectif général de ce travail est de progresser vers la **compréhension des données d'observation de la Terre à grande échelle par des méthodes semi-supervisées**. En effet, il n'est pas possible aujourd'hui d'obtenir des données étiquetées à grande échelle et une solution envisageable pour améliorer les capacités de généralisation des modèles est d'exploiter l'abondance de données non étiquetées disponibles dans le domaine, en intégrant toutes les connaissances qui leur sont intrinsèquement imprimées. À cette fin, nous abordons le problème principalement sous deux angles :

- ▶ **Les données.** La segmentation sémantique semi-supervisée est une tâche relativement nouvelle. Par conséquent, avant ce travail et à notre connaissance, il n'existait aucun jeu de données permettant de comparer équitablement les méthodes semi-supervisées de segmentation sémantique dans le domaine.
- ▶ **Les algorithmes.** Comment intégrer des données non étiquetées dans les modèles de segmentation sémantique et de classification de scènes ? Faut-il aborder le problème d'un point de vue discriminatif ou génératif ?

Enfin, un point important et peut-être un concept clé à garder à l'esprit comme objectif de cette thèse est la **généralisation**. En effet, il n'est pas possible, à ce jour, d'obtenir des données étiquetées à l'échelle mondiale. Comment généraliser à travers des lieux géographiques lorsque les étiquettes ne sont disponibles qu'à partir d'un lieu spécifique ? Même s'il n'y a pas de chapitre spécialement consacré à la généralisation, ce concept est constamment mentionné tout au long de cet ouvrage.

Apprentissage semi-supervisé pour la compréhension des données d'observation de la Terre à large échelle

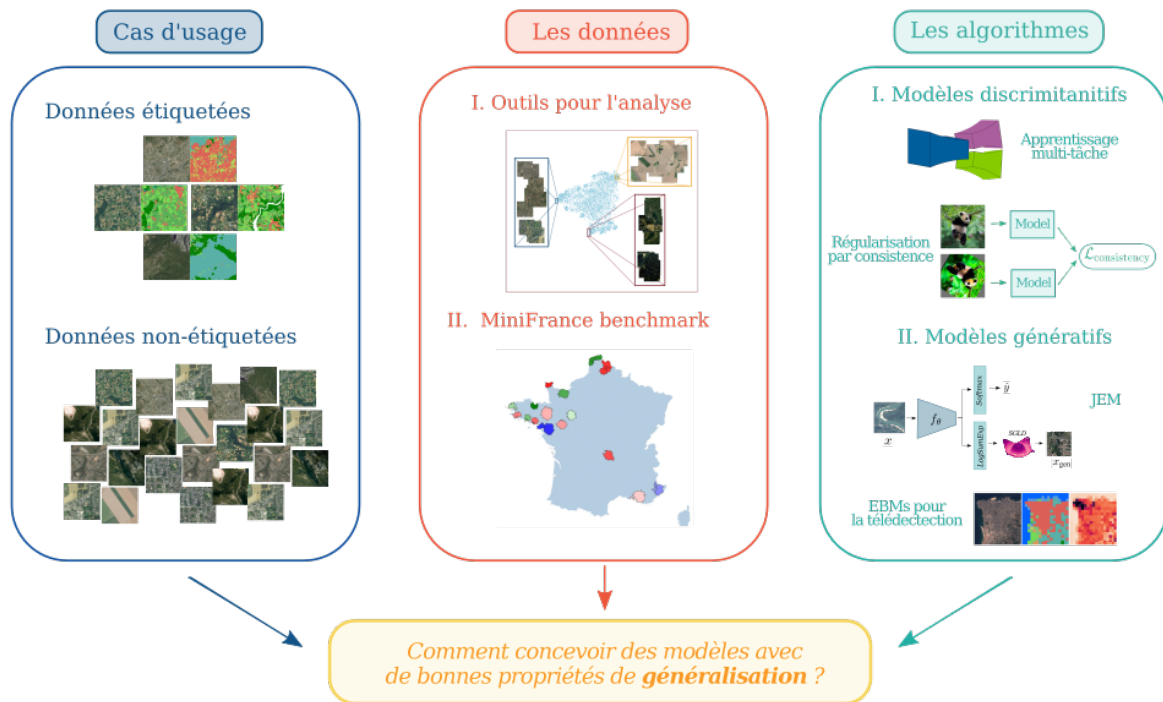


Figure 1 – Ce travail porte sur l’apprentissage semi-supervisé pour la compréhension des données d’observation de la Terre. Le but est de tirer parti de quelques données étiquetées et de grandes quantités d’échantillons non étiquetés pour entraîner des modèles pour la cartographie de l’utilisation des sols. Nous abordons le sujet sous deux angles : les données, en fournissant des outils pour l’analyse des données et une nouvelle base des données pour la segmentation sémantique semi-supervisée –MiniFrance ; et les algorithmes, en étudiant les méthodes discriminatives et génératives pour construire des modèles avec de bonnes capacités de généralisation.

Contributions

Chapitre 2

Le potentiel de l'apprentissage semi-supervisé pour l'observation de la Terre

Ce chapitre présente une analyse des jeux de données d'observation de la Terre existants d'un point de vue critique : modèlent-ils les applications de télédétection réelles ? Qu'attendons-nous d'un *bon* jeu de données ? Dans les applications générales d'OT, on aimerait entraîner un modèle qui généralise correctement à différents endroits géographiques. De plus, on a généralement accès à très peu de données étiquetées, alors que de nombreuses données non étiquetées sont disponibles. Par conséquent, les jeux de données doivent simuler ces situations pour être considérés comme un repère d'évaluation approprié et fiable.

En outre, nous étudions les capacités d'apprentissage des approches supervisées actuelles dans différents contextes : sur des jeux de données à petite échelle et dans une configuration à grande échelle et à plusieurs endroits. Nos expériences montrent que les réseaux de segmentation sémantique supervisés courants ont des problèmes de généralisation dans un contexte à grande échelle. Il existe donc une opportunité pour de nouveaux paradigmes d'apprentissage : apprentissage semi-supervisé, apprentissage faiblement supervisé, apprentissage actif, etc. Dans ce travail, nous nous concentrons sur les techniques d'apprentissage semi-supervisé, car la pléthore de données d'OT non étiquetées disponibles devrait être exploitée pour développer des modèles robustes et génériques.

Enfin, ce chapitre présente la suite MiniFrance, un nouveau jeu de données à grande échelle conçu pour la segmentation sémantique semi-supervisée dans l'observation de la Terre. MiniFrance possède des propriétés sans précédent, la diversité des paysages et des scènes reflète la complexité de la réalité. Par-dessus tout, il a été soigneusement conçu pour l'apprentissage semi-supervisé, en incluant des données étiquetées et non étiquetées dans sa partition d'entraînement et en recréant un cadre d'application réaliste, ce qui rend MiniFrance unique. En plus de cette base de données, nous présentons une analyse complète des données en termes de similarité d'apparence et de représentativité, montrant que MiniFrance est bien adapté pour traiter le problème semi-supervisé.

Chapitre 3

Les méthodes discriminatives pour l'apprentissage semi-supervisé

Ce chapitre est consacré à l'étude de l'apprentissage semi-supervisé dans une perspective discriminative. Dans ce contexte, nous étudions deux familles d'algorithmes : les méthodes d'apprentissage multi-tâche et les approches basées sur la régularisation de la cohérence.

Dans le cadre de l'apprentissage multi-tâche, nous présentons des réseaux de neurones profonds multi-tâche pour effectuer la segmentation sémantique semi-supervisée. En particulier, nous proposons BerundaNet – une extension simple des architectures classiques d'encodeur-décodeur – qui s'avère très efficace dans la tâche semi-supervisée. Avec ces architectures, nous explorons les fonctions de perte auxiliaires non supervisées à utiliser en parallèle avec la segmentation sémantique. En particulier, nous proposons la perte k-means relaxée pour effectuer une segmentation d'image non supervisée.

Nos expériences sur trois jeux de données disponibles publiquement pour la segmentation sémantique semi-supervisée ont montré que nous pouvons bénéficier de données non étiquetées pendant le processus d'apprentissage pour améliorer les cartes de segmentation sémantique. En effet, les approches semi-supervisées permettent de générer des prédictions plus fines et plus homogènes. Nous avons également observé qu'une architecture simple comme BerundaNet-late avec un backbone approprié comme U-Net est suffisante pour améliorer les performances de segmentation.

Néanmoins, le problème de l'apprentissage semi-supervisé n'est pas encore résolu. Nous avons vu que ces approches multi-tâches peuvent améliorer les résultats de la segmentation sémantique, mais ce n'est pas toujours le cas. Dans une approche multi-tâche comme celles présentées dans ce chapitre, il faut faire attention au choix de l'architecture et de la tâche auxiliaire à réaliser en parallèle. En outre, il existe d'autres façons de résoudre le problème de la semi-supervision. Par exemple, on peut développer des modèles génératifs pour apprendre la distribution intrinsèque des données à partir d'exemples étiquetés et non étiquetés et utiliser cette information avec les étiquettes pour améliorer la segmentation. Une autre possibilité consiste à utiliser des méthodes de pseudo-étiquetage qui propagent les étiquettes des exemples annotés à travers les exemples non annotés, sur la base d'un critère de confiance, afin d'élargir les données d'apprentissage disponibles.

La deuxième partie du chapitre explore les méthodes basées sur le principe de régu-

larisation de la cohérence. La régularisation de la cohérence est l'une des techniques les plus largement appliquées dans les algorithmes actuels de classification semi-supervisée. Elle applique l'idée qu'un modèle doit produire des prédictions similaires pour des exemples sémantiquement similaires. Nous présentons un cadre théorique basé sur la minimisation du risque vicinal pour justifier l'utilisation de la régularisation de cohérence. Ensuite, nous présentons FixMatch, la méthode de pointe actuelle pour la classification semi-supervisée en vision par ordinateur. Enfin, nous avons réalisé des expériences sur deux jeux de données d'observation de la Terre accessibles au public pour la classification de scènes.

Nos expériences démontrent l'efficacité et la transférabilité de méthodes telles que FixMatch au domaine de l'observation de la Terre. De plus, elles montrent que la régularisation de la cohérence, avec le bon ensemble de transformations de données, améliore la robustesse des modèles par rapport aux changements de domaine, ce qui est une caractéristique souhaitable dans les applications d'observation de la Terre.

Chapitre 4

Les méthodes génératives pour l'apprentissage semi-supervisé

Ce chapitre étudie l'apprentissage semi-supervisé d'un point de vue génératif. À cette fin, nous définissons d'abord ce que sont les modèles génératifs et expliquons brièvement les grands principes des différents cadres génératifs profonds.

Malgré certains inconvénients, les modèles basés sur l'énergie présentent plusieurs avantages par rapport aux autres modèles génératifs. Ils capturent toutes les informations sur les entrées uniquement à travers une valeur scalaire, *l'énergie*. L'estimation de l'énergie par le biais d'un réseau de neurones permet de modéliser des distributions complexes, ce qui rend les EBM très intéressants pour plusieurs applications, notamment la génération, la détection d'exemples hors-distribution, etc. De plus, leur simplicité permet d'intégrer naturellement l'information de l'étiquette dans le modèle, en estimant une fonction d'énergie jointe $E(x, y)$, avec très peu de changements sur l'architecture du réseau de neurones à utiliser, et aucun changement sur le processus d'optimisation.

Dans ce contexte, nous considérons une méthode récente qui permet d'entraîner les réseaux de neurones à effectuer conjointement la classification et la génération d'images. Nous appliquons ce modèle aux données de télédétection. En réinterprétant les sorties

d'un réseau de classification, le modèle conjoint basé sur l'énergie (JEM) exprime la distribution jointe des paires image-étiquette comme un modèle basé sur l'énergie. En pratique, il nous permet d'entraîner un classifieur robuste et d'estimer la distribution sous-jacente des données, simultanément. De plus, ce modèle hybride est bien adapté et s'étend naturellement à l'apprentissage semi-supervisé.

Cette application séminale de JEM aux données d'observation de la Terre a conduit à plusieurs conclusions importantes. Tout d'abord, dans les jeux de données à petite échelle comme EuroSAT, nous observons que JEM est un classifieur puissant dont les performances sont comparables à celles des méthodes de pointe. Plus intéressant encore, dans le cadre semi-supervisé, lorsque très peu d'exemples étiquetés sont disponibles, JEM est supérieur à un réseau supervisé standard, tant en termes de scores de classification que de robustesse (c'est-à-dire, il est mieux calibré). Deuxièmement, avec des jeux de données plus réalistes et à grande échelle comme So2Sat, JEM présente des propriétés de généralisation exceptionnelles, avec de meilleures performances que les classifieurs habituels dans les contextes supervisé et semi-supervisé. Cependant, les travaux futurs pourraient se concentrer sur l'intégration dans JEM des mécanismes de FixMatch spécialement conçus pour l'apprentissage semi-supervisé, à savoir les techniques d'augmentation des données, le pseudo-étiquetage ou les stratégies de régularisation de la cohérence. Le défi consiste à augmenter les données de manière réaliste, et l'estimation de la distribution fournie par JEM pourrait être un atout à cet égard.

Nous avons également démontré que JEM est capable d'estimer correctement la distribution des données, ce qui nous permet de générer des images fidèles et diverses. L'estimation de la distribution des données permet au modèle de détecter les échantillons hors distribution et donc de décider s'il peut être utilisé de manière fiable dans un nouveau domaine. Cela donne à JEM la capacité de classer les zones non vues avec une carte de confiance basée sur la log-vraisemblance estimée par le modèle.

Malgré les limites et les problèmes de convergence de l'entraînement des modèles basés sur l'énergie, nous avons montré par nos expériences plusieurs applications intéressantes en télédétection pour ce type de modèle hybride discriminatif-génératif, comme l'apprentissage semi-supervisé, la détection de la non-répartition ou la génération de nouvelles données synthétiques réalistes. C'est un point de départ pour ouvrir la voie aux applications réelles de demain.

Enfin, nous présentons une extension théorique de ce modèle d'énergie à la segmentation sémantique. Cependant, l'application pratique de cette extension n'en est qu'à ses

débuts, elle est donc laissée ouverte comme perspective pour de travaux futurs.

Publications

Le travail présenté dans ce manuscrit a donné lieu à des publications dans des revues à comité de lecture, comme nous le détaillons ci-dessous :

Journal articles

- [J1] J. Castillo-Navarro, B. Le Saux, A. Boulch, and S. Lefèvre, « Energy-based models in Earth observation: from generation to semi-supervised learning », *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, 2022.
- [J2] J. Castillo-Navarro, B. Le Saux, A. Boulch, N. Audebert, and S. Lefèvre, « Semi-supervised semantic segmentation in Earth observation: the MiniFrance suite, dataset analysis and multi-task network study », *Machine Learning*, pp. 1–36, 2021 (cit. on pp. 60, 147).

Conference articles

- [C1] J. Castillo-Navarro, N. Audebert, A. Boulch, B. Le Saux, and S. Lefèvre, « What data are needed for semantic segmentation in Earth observation? », in *2019 Joint Urban Remote Sensing Event (JURSE)*, IEEE, 2019.
- [C2] J. Castillo-Navarro, B. Le Saux, A. Boulch, and S. Lefèvre, « Réseaux de neurones semi-supervisés pour la segmentation sémantique en télédétection », in *Colloque GRETSI*, 2019.
- [C3] J. Castillo-Navarro, B. Le Saux, A. Boulch, and S. Lefèvre, « On auxiliary losses for semi-supervised semantic segmentation », in *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery Workshops (ECML-PKDD W) - MACLEAN*, 2020.
- [C4] J. Castillo-Navarro, B. Le Saux, A. Boulch, and S. Lefèvre, « Classification and generation of Earth observation images using a joint energy-based model », in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2021.

- [C5] J. Castillo-Navarro, B. Le Saux, A. Boulch, and S. Lefèvre, « Energy-based models for Earth observation applications », in *Proceedings of the International Conference on Learning Representations - Energy Based Models Workshop - (ICLR-W)*, 2021.

Others

- [O1] R. Hänsch, C. Persello, G. Vivone, J. Castillo-Navarro, A. Boulch, S. Lefèvre, and B. Le Saux, « 2022 IEEE GRSS Data fusion contest: semi-supervised learning [technical committees] », to appear in *IEEE Geoscience and Remote Sensing Magazine*, 2022.

INTRODUCTION

Contents

Context	22
Objectives	25
Organization of this document	27
Publications	28

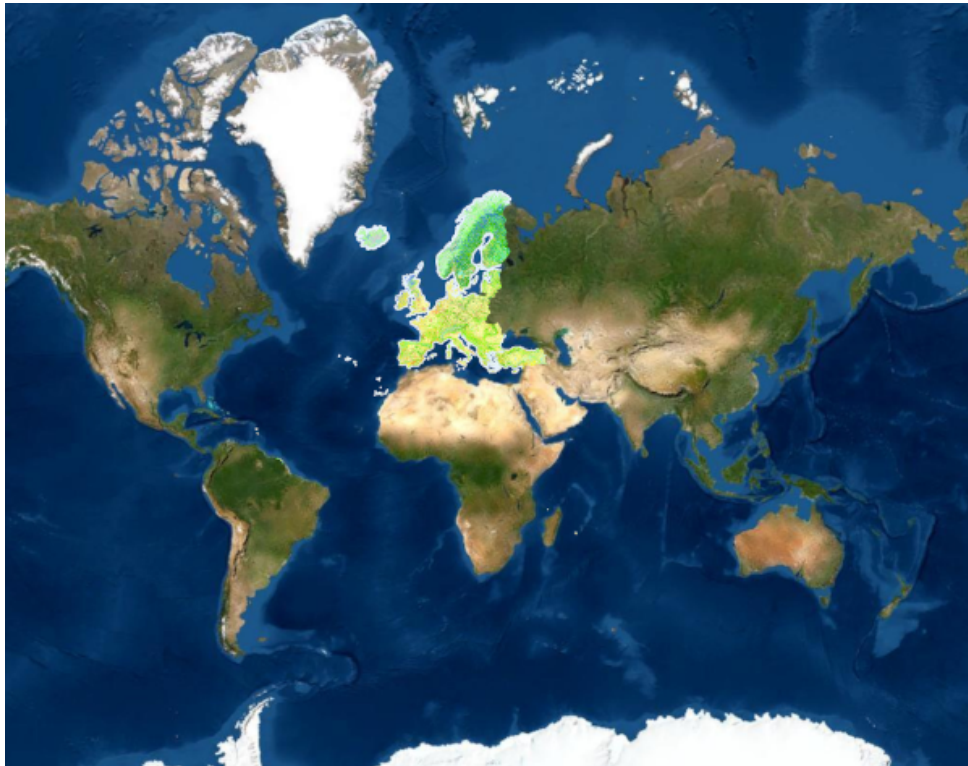


Figure 2 – Global imagery of the Earth, together with the European Corine Land Cover map. Even using one of the largest label sources available, labeled images are only a tiny fraction of all imagery available, furthermore taking into account the revisiting of the satellites which image the same zone of the globe repeatedly. How can we leverage the massive amounts of unlabeled data available? World Imagery by Esri. Corine land cover from the Copernicus program.

violet (UV), infrared (IR) or microwave radiation. The fact that satellites measure in different spectral wavelengths is used for obtaining information about the objects, features or properties under study. Different sensors give rise to different kinds of remote sensing imagery, including multi-spectral, hyper-spectral, SAR, LiDAR, etc.

Even though the first images of Earth from space were taken in 1946 from a camera attached to a rocket over New Mexico, it was the launching of Sputnik 1 –in 1957– that marked the beginning of the satellite remote sensing era. Sputnik 1 transmitted radio signals that were received on Earth. It was in 1959, that the Explorer 6 took the first photos of our planet Earth from a satellite. Ever since, efforts have multiplied and as of today, more than 150 Earth observation satellites are currently in orbit. They not only provide spectacular views of our planet, but also significant scientific insights at a global scale. For instance, Sentinel-2 satellites image the entire Earth in only 5 days.

These satellite constellations have provided a deluge of Earth observation data, with storage volumes beyond dozens of petabytes⁵, acquiring more than hundreds of terabytes of data per day. For instance, Sentinel-2 satellites alone provide more than 20 TB of data per day [2, 3]. Interpreting and understanding satellite imagery require a certain domain expertise, combining knowledge of the physics of the sensors and knowledge of the considered application. However, exploiting these gargantuan amounts of data is not humanly possible. Automatic Earth observation image analysis seems to be the way to extract the insightful information contained in those data.

In the last decade, deep learning techniques –and the consequential growth of computing power– have transformed the fields of computer vision and image processing. More recently, deep learning has shown increasing evidence of the potential to address problems in Earth and climate sciences as well [2, 4, 5]. These techniques represent promising tools to build new data-driven models for Earth observation.

Unfortunately, most of the learning algorithms developed to date heavily rely on the availability of massive annotated image databases. In general, labeled data are hard to obtain, necessitating resources, time and expert knowledge. Moreover, there is no efficient way to deliver humanly annotated labels for the immensity of EO data available. On the other hand, raw data –without labels– are abundant, especially in remote sensing where satellites generate data continuously, as illustrated by Fig. 2. Because of this, we are convinced that semi-supervised methods –which leverage unlabeled data to help on the learning process– will be essential to push further the generalization

5. 1 petabyte = 10^{15} bytes.

capacities of the models.

Therefore, the main goal of this thesis is to advance in the road toward large-scale automated cartography. To this end, we work on the analysis and development of new semi-supervised methods that leverage the abundance of unlabeled remote sensing imagery and make our best to go beyond current state-of-the-art fully supervised models.

This work is then at the crossroads of three domains: remote sensing for Earth observation, machine learning and computer vision. Remote sensing involves all the techniques of observing and analyzing objects from a distance. When it comes to Earth observation, remote sensing often refers to information collected through Earth observation satellites, but it may also include airborne or UAV⁶ collected data. Among all the possible applications of Earth observation, in this work we focus on land use and land cover mapping of the Earth. Machine learning is the subfield of computer science that studies algorithms that can learn and improve automatically through experience and by the use of data. In particular, this study involves deep learning, a specific branch of machine learning that uses neural networks at its core. In particular, in this thesis we study and develop classification and segmentation algorithms, since they are related to the land use and land cover mapping problem. Finally, computer vision comprises all the techniques developed for automatic interpretation of images. Similarly to the human vision system, computer vision aims to extract useful information from images, addressing tasks such as image classification, object detection, segmentation, just to name a few examples. In this work we exploit these techniques for information extraction, analysis and understanding of Earth observation images.

In this way, this thesis aims to develop deep learning and computer vision techniques to serve remote sensing and Earth observation applications, using EO imagery.

6. Unmanned aerial vehicles, also known as drones.

Objectives

The **general objective** of this work is to progress toward **large-scale cartography and Earth observation data understanding through semi-supervised methods**. Indeed, today it is not possible to obtain labeled data at a large-scale and a feasible solution to improve the generalization capacities of our models consists in leveraging the abundance of unlabeled data in the field, integrating all the knowledge intrinsically imprinted on them. To this end, we address the problem from mainly two angles:

- ▶ **The data.** Semi-supervised semantic segmentation is a relatively new task. Therefore, before this work and to the best of our knowledge, there was no dataset that allowed us to fairly compare semi-supervised methods for semantic segmentation in the field.
- ▶ **The algorithms.** How to integrate unlabeled data into semantic segmentation and scene classification models? Should we address the problem from a discriminative or a generative perspective?

Finally, an important point and key concept to keep in mind as a goal for this thesis is **generalization**. Indeed, it is not possible⁷ to obtain labeled data at a global-scale. How to generalize across geographic locations when labels are available only from a specific location? Even if there is no chapter especially devoted to generalization, this concept is constantly mentioned throughout this work. Fig. 4 summarizes our main goal, the use-case and our approaches.

In consequence, this manuscript addresses the following research questions:

1. **Are unlabeled data useful to perform semantic mapping?** Is supervised learning with existing datasets sufficient to achieve large-scale mapping? How do we measure the information contained in a dataset? In our experiments, we show the limits and the potential of unlabeled data depending on the complexity of the task. We also propose new tools to analyse multi-location datasets, as it is usually the case in EO.
2. **Can we adapt discriminative models to perform semi-supervised learning?** How can we integrate unlabeled data into the training process of neural networks? Do they improve their performance? We develop multi-task methods for semi-supervised semantic segmentation and study techniques based on consistency

7. as of today.

Semi-supervised learning for large-scale EO data understanding

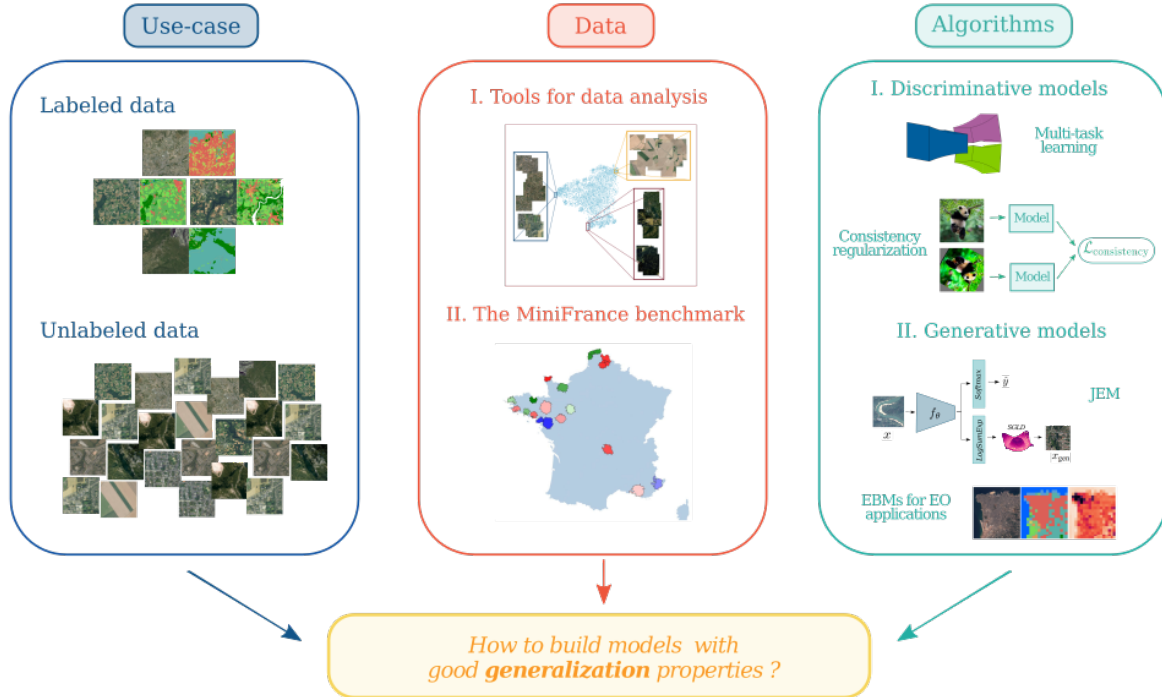


Figure 4 – This work focuses on semi-supervised learning for Earth observation data understanding. Our goal is to leverage few labeled data and large amounts of unlabeled samples to train models for land use and land cover mapping. We tackle the subject from two angles: the data, providing tools for data analysis and a new benchmark for semi-supervised semantic segmentation –MiniFrance; and the algorithms, investigating discriminative and generative methods to build models with good generalization capacities.

regularization for semi-supervised scene classification. We show that integrating unlabeled data improves the performance on these tasks and yields better generalization on unseen geographic areas.

3. **Should we use generative models for semi-supervised learning?** How can we integrate label information into a traditionally unsupervised learning process? Is the estimation of data distribution useful for classification purposes? We study generative models like energy-based models –in particular a joint energy-based model (JEM)– in EO use-cases. We show that generative models are clearly a promising direction, providing more interpretable models with good generalization capacities. However, they are still at an early stage, due to task learning complexity.

Organization of this document

In order to establish the basis to fully understand the content and contributions of this thesis work, Chapter 1 defines the main concepts that are recurrently mentioned in this manuscript, summarizes the main tools that are used along this study, and gives an overview of previous research directly related to this subject.

Our scientific contributions are presented in Chapters 2, 3 and 4. Chapter 2 performs a critical analysis to current supervised learning in EO and brings out other possibilities to achieve good generalization at a large-scale, such as semi-supervised learning. In this regard, the MiniFrance suite is presented, the first dataset especially designed to benchmark semi-supervised algorithms in the field. Moreover, tools for representativeness assessment of multi-location data are developed and applied to MiniFrance to have a thorough understanding of this dataset.

Chapter 3 explores semi-supervised learning in Earth observation with discriminative models. The first part of this chapter develops multi-task approaches for semi-supervised semantic segmentation. Several experiments are performed to assess their performances and capacities. The second part of this chapter is devoted to the study of methods based on the consistency regularization principle. In particular, state-of-the-art methods for semi-supervised classification in computer vision are evaluated on remote sensing data, demonstrating their potential in the field.

The use of generative models for semi-supervised learning and various Earth observation applications is investigated in Chapter 4. In particular, this chapter presents a thorough study of a recent energy-based framework, showing that this kind of models can be successfully applied to Earth observation data. Moreover, the conducted research shows the capabilities of such versatile models for several applications of high interest in Earth observation.

Finally, the [Conclusion](#) chapter closes this manuscript, summarizing the main contributions and conclusions of this work. Moreover, it presents some perspectives on future projects (either short or long term) to continue the research on semi-supervised learning in the field of remote sensing.

Publications

The work presented in this manuscript has given rise to publications in peer reviewed venues, as we detail below:

Journal articles

- [J1] J. Castillo-Navarro, B. Le Saux, A. Boulch, and S. Lefèvre, « Energy-based models in Earth observation: from generation to semi-supervised learning », *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, 2022.
- [J2] J. Castillo-Navarro, B. Le Saux, A. Boulch, N. Audebert, and S. Lefèvre, « Semi-supervised semantic segmentation in Earth observation: the MiniFrance suite, dataset analysis and multi-task network study », *Machine Learning*, pp. 1–36, 2021 (cit. on pp. 60, 147).

Conference articles

- [C1] J. Castillo-Navarro, N. Audebert, A. Boulch, B. Le Saux, and S. Lefèvre, « What data are needed for semantic segmentation in Earth observation? », in *2019 Joint Urban Remote Sensing Event (JURSE)*, IEEE, 2019.
- [C2] J. Castillo-Navarro, B. Le Saux, A. Boulch, and S. Lefèvre, « Réseaux de neurones semi-supervisés pour la segmentation sémantique en télédétection », in *Colloque GRETSI*, 2019.
- [C3] J. Castillo-Navarro, B. Le Saux, A. Boulch, and S. Lefèvre, « On auxiliary losses for semi-supervised semantic segmentation », in *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery Workshops (ECML-PKDD W) - MACLEAN*, 2020.
- [C4] J. Castillo-Navarro, B. Le Saux, A. Boulch, and S. Lefèvre, « Classification and generation of Earth observation images using a joint energy-based model », in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2021.
- [C5] J. Castillo-Navarro, B. Le Saux, A. Boulch, and S. Lefèvre, « Energy-based models for Earth observation applications », in *Proceedings of the International Conference on Learning Representations - Energy Based Models Workshop - (ICLR-W)*, 2021.

Others

- [O1] R. Hänsch, C. Persello, G. Vivone, J. Castillo-Navarro, A. Boulch, S. Lefèvre, and B. Le Saux, « 2022 IEEE GRSS Data fusion contest: semi-supervised learning [technical committees] », *to appear in IEEE Geoscience and Remote Sensing Magazine*, 2022.

RELATED WORK

Contents

Chapter summary	32
1.1 Semantic segmentation	32
1.2 Learning paradigms	36
1.2.1 Supervised learning	37
1.2.2 Unsupervised learning	38
1.2.3 Semi-supervised learning	38
1.2.4 Discriminative vs. generative models	40
1.3 Deep learning	41
1.3.1 Brief history of deep neural networks	41
1.3.2 The multilayer perceptron	43
1.3.3 Convolutional neural networks	46
1.3.4 Fully convolutional networks	48
1.3.5 Deep semi-supervised learning	50
1.3.6 Deep learning in Earth observation	52

Chapter summary

This work is at the intersection of different fields of research, like computer vision, deep learning and Earth observation applications and is based on many previous works developed by researchers over decades.

This chapter attempts to present the fundamental concepts and frameworks on which this thesis relies. First, we define what semantic segmentation is and establish its connection to cartography in remote sensing. Secondly, we present different machine learning paradigms according to the nature of available data or the kind of task to perform. Finally, we outline the principles of deep learning and delve into neural networks for image understanding; we present the essentials of deep semi-supervised learning and conclude the chapter with a brief overview of deep learning for Earth observation applications.

1.1 Semantic segmentation

The human brain has a natural ability for pattern recognition. Indeed, humans are able to quickly identify structures and shapes, organizing information into meaningful parts. Our vision system has evolved in such a way that it is able to enhance contours, distinguish features, get a visual perception and recognize objects [15].

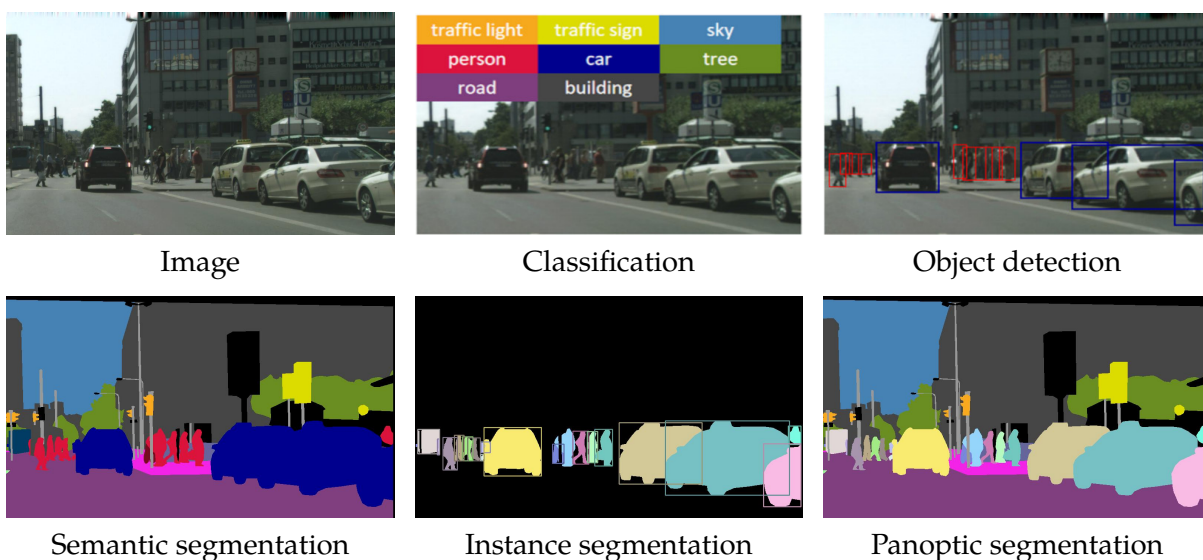


Figure 1.1 – Example of different vision tasks. Figure borrowed from [16].

Computer scientists have tried to mimic these human abilities with computers. Thereby, modern computer vision tasks have been defined, including image classification, object detection and different kinds of image segmentation, as shown in Fig. 1.1. In particular, *image segmentation* corresponds to the task of dividing an image into non-overlapping meaningful entities, called segments. A segment is a set of pixels that share some characteristics (texture, color, etc.). Image segmentation has been a well-studied problem in computer vision and different approaches have been developed, like clustering-based approaches, superpixel segmentation, etc.

Yet, the previous definition of segmentation is somehow ambiguous. We would expect that a segmentation algorithm decomposes an image into objects or meaningful segments. However, can we define precisely what makes a “meaningful segment”? It could be an item, like a bottle, a table or a rabbit; or it could be a color; or a texture, like wood, rocks, etc. Fig. 1.2 illustrates how human perception of “meaningful segments” can vary. Indeed, it shows an example where different human annotators have different ways of interpreting the same scene.

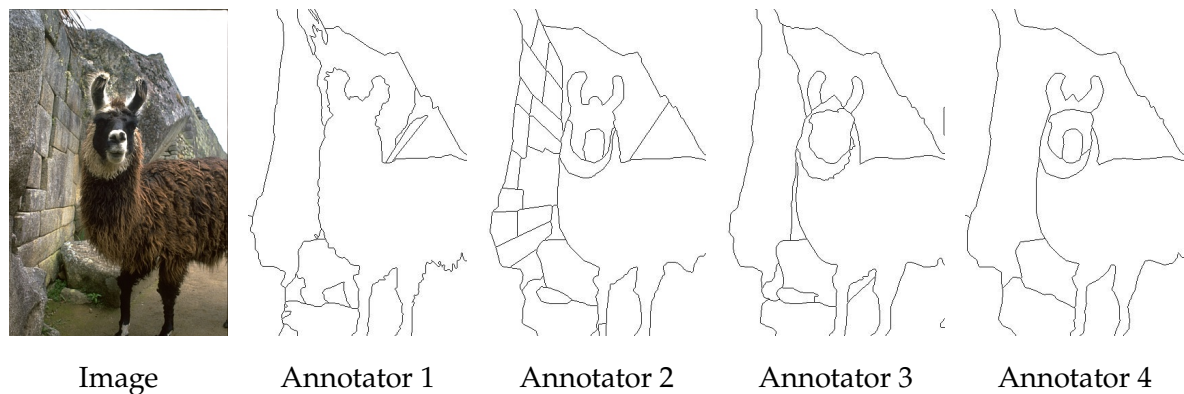


Figure 1.2 – Image from Berkeley segmentation dataset [17] with hand labels produced by four different human annotators. These images reveal the variety of human perception and reflect the ambiguity in the previous segmentation definition.

Semantic segmentation refines this definition. The idea of semantic segmentation is to divide an image into *semantically meaningful segments*. In other words, it consists in the process of assigning a class label to every pixel on an image. It is a relevant and challenging task in computer vision because it implies understanding the context of a scene or an image, and has been extensively studied because of its multiple, high-potential applications in several domains, such as autonomous driving, computer aided

diagnosis, robot vision and understanding, etc.

The main goal of this thesis work is to progress toward large-scale cartography, namely, land use and land cover mapping of remote sensing imagery. Land use and land cover maps, both represent spatial information of different classes¹ of physical coverage or use of the Earth’s surfaces. If we represent the surface of our planet as an image, land use and land cover maps correspond exactly to the semantic segmentation of the image, which explains our interest in this particular task.

Even if segmentation seems to be a very natural skill for humans [18], it is a very challenging task in computer vision that has been extensively studied over the last years [16, 18–21]. Several methods have been developed over the years to fulfill semantic segmentation [18]. However, the most successful algorithms today are based on deep neural networks. The breakthrough of Convolutional Neural Networks (CNN) and, more specifically, Fully Convolutional Neural Networks (FCN) revolutionized the way of obtaining dense predictions [22, 23]. These new architectures allow us to generate segmentation maps for images of any size and most of the subsequent state-of-the-art approaches adopted this paradigm.

Evaluation metrics

In order to measure the actual contribution and the validity of any learning algorithm, we need to somehow evaluate its performance at its learning task.

Several evaluation metrics exist in the field of semantic segmentation and, depending on the context of the problem, one should give more importance to some metrics over the others.

In what follows we present some of the most usual metrics that are currently employed to measure the performance of semantic segmentation algorithms.

For a classifier, let i be one class of interest. Let TP be the set of true positive examples for class i (examples from class i that have been correctly classified), TN the set of true negative examples for class i (examples from a class $j \neq i$ that have not been classified as i), FP the set of false positive examples for class i (examples from a class $j \neq i$ that have been wrongly classified as i) and FN the set of false negative examples for class i (examples from class i that have been wrongly classified as class $j \neq i$).

1. In the case of land cover, classes correspond to physical coverage of the Earth’s surface (forests, grasslands, lakes, etc). Land use, on the other hand, refers to classes related to the arrangements or activities that humans undertake on a certain land (recreation, agriculture, etc).

We define the following performance evaluation metrics, with respect to class i :

Accuracy: It is the simplest metric. It computes the ratio between the amount of properly classified pixels and the total number of them.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} .$$

Accuracy measures the percentage of well classified data.

Intersection over Union (IoU): It is defined as the ratio between the intersection and the union of the ground truth and the obtained predictions. This is

$$\text{IoU} = \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}} .$$

Precision: This metric is defined as the ratio between the number of true positive examples and the total number of examples inferred to belong to the class.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} .$$

Precision can be seen as an accuracy measure for the “positive” examples (elements that truly belong to class i).

Recall: It is the quotient between the number of true positive examples and the number of examples that really belong to the class.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} .$$

This metric can be thought as the ability of the classifier to correctly identify the relevant cases: to obtain a high recall score we might maximize the number of true positive examples and, at the same time, minimize the number on false negative examples.

F_1 score: It is also known as the Sørensen-Dice coefficient. This metric is computed as the harmonic mean of precision and recall:

$$F_1 = 2 \cdot \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} ,$$

this formula can be rewritten as:

$$F_1 = \frac{2TP}{2TP + FP + FN} .$$

When we are facing an imbalanced classification problem, i.e., a problem where one class represents the majority of the dataset points, accuracy is not an adequate metric. For example, let us consider a dataset composed of 95% of background and 5% of an object, and a classifier that always predicts “background”. Then, the accuracy of the classifier is 95 %, but a classifier always predicting the same class is not very useful.

In the context of the previous example, the F_1 score or the IoU may be better metric alternatives, since they are not biased in favor of a predominant class. Even if the accuracy of the classifier on the example is 95 %, its F_1 score is 0.

To summarize, there exist different evaluation metrics measuring the performance of a classifier, and we can choose the ones that are more adequate to our task. Nevertheless, to keep scientific rigor, it is important to provide all the possible metrics for a proposed method, avoiding redundancy.

1.2 Learning paradigms

Machine learning is a subfield of computer science, whose goal is to develop algorithms that allow computers to “learn”.

In machine learning, instead of teaching a computer a massive list of rules to solve the problem, we give it a model with which it can evaluate examples. Additionally, we provide a small set of instructions to modify the model when it makes a mistake. We expect that, over time, a well-suited model would be able to satisfactorily solve the problem. Fig. 1.3 illustrates the fundamental difference between classical programming and machine learning.

According to Tom Mitchell [24], a computer program is said to *learn* from **experience E** with respect to some class of **tasks T** and **performance measure P**, if its performance at tasks in T, as measured by P, improves with experience E.

From the previous definition, the key ingredients that any machine learning algorithm needs to work are: a well-defined task T, experience E to learn from (usually, a collection of data samples) and a measure of performance P to adapt itself.

For example, if we want to develop a classification algorithm, the task T to perform

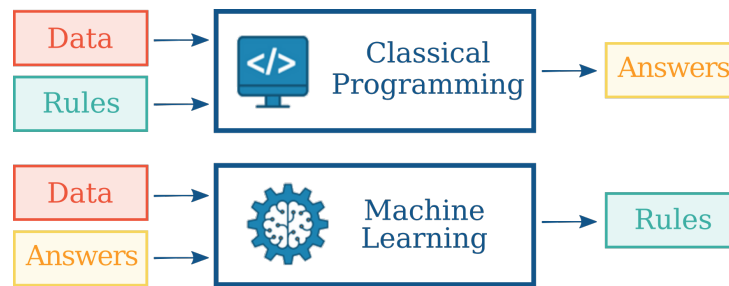


Figure 1.3 – Classical programming vs. machine learning.

is assigning a class to a sample; the experience E would be the data, given as pairs (sample, label); and the performance measure P can be the misclassification rate. Similarly, for an object clustering algorithm, the task is grouping objects according to similar characteristics, the experience would be the set of objects and the performance measure, the quality of groups.

Depending on the kind of task to develop or the nature of data available for the algorithm to learn, machine learning techniques can be grouped into different families. Traditionally, there exist two main types of tasks in machine learning: supervised learning and unsupervised learning. From them, we can derive other definitions: semi-supervised learning [25], weakly-supervised learning [26], self-supervised learning [27].

1.2.1 Supervised learning

Let $\mathbf{X} = \{x_i\}_{i=1}^n$ be a set of n examples (data points), where $x_i \in \mathcal{X}, \forall i \in \{1, \dots, n\}$. Let $\mathbf{Y} = \{y_i\}_{i=1}^n$, with $y_i \in \mathcal{Y}, \forall i \in \{1, \dots, n\}$ targets corresponding to samples in \mathbf{X} . Usually, it is assumed that pairs (x_i, y_i) are i.i.d samples following $P(x, y)$, a probability measure defined over $\mathcal{X} \times \mathcal{Y}$.

The goal of **supervised learning** is to find a map $x \mapsto y$, given the training set previously defined. The task is well defined, since this mapping can be evaluated through its predictive performance on test samples. When labels y are continuous ($\mathcal{Y} = \mathbb{R}^d$) we talk about a regression problem, for instance, predicting weather forecast. On the other hand, when targets y take values on a finite set we refer to a classification problem, for example predicting whether e-mails are spam or not.

1.2.2 Unsupervised learning

In an *unsupervised learning* problem, one only has access to a set $\mathbf{X} = \{x_i\}_{i=1}^n$ of n samples, where $x_i \in \mathcal{X}, \forall i \in \{1, \dots, n\}$. x_i samples are supposed to be i.i.d., following a distribution $P(x)$ defined over \mathcal{X} . The main difference with respect to supervised learning is that no target information is available. In this settings, the goal is to find interesting properties or structure in data \mathbf{X} . Usual unsupervised learning applications are dimensionality reduction, clustering, quantile estimation. Estimating the density which is likely to have generated \mathbf{X} is also an unsupervised task.

Lately, a new kind of unsupervised algorithms have emerged with astounding results in representation learning: self-supervised learning. Self-supervised learning methods build a supervised task from completely unlabeled data by producing labels from the data themselves. Today, self-supervised learning and, especially, contrastive learning methods are a very active topic of research, with the state-of-the-art methods for learning representations [28, 29].

1.2.3 Semi-supervised learning

Semi-supervised learning [25] refers to all the techniques that are halfway between supervised and unsupervised learning. In these settings, available data can be divided into two parts: (i) a labeled set where data samples and their corresponding targets are provided $D_\ell = \{(x_i, y_i)\}_{i=1}^n$, where (x_i, y_i) pairs satisfy the same hypothesis as in the supervised setting (Section 1.2.1); and (ii) an unlabeled set for which only raw data are available, $D_u = \{u_i\}_{i=1}^m$, and we assume $\{u_i\}_{i=1}^m$ i.i.d, following $P(x)$, with $P(x)$ the marginal distribution of $P(x, y)$. Usually we also consider that $n \ll m$.

The key idea behind semi-supervised learning is to learn a representation function (that maps a data point to its target) from labeled data as in the supervised approach, but using the available unlabeled data to leverage information about structure of these data to help during the learning process. Fig. 1.4 illustrates this kind of situations. This is a much realistic and compelling approach than supervised learning, since in real-life applications annotated data is difficult to procure –even harder in the context of semantic segmentation, since one needs pixel-wise labels– while raw data are plentiful.

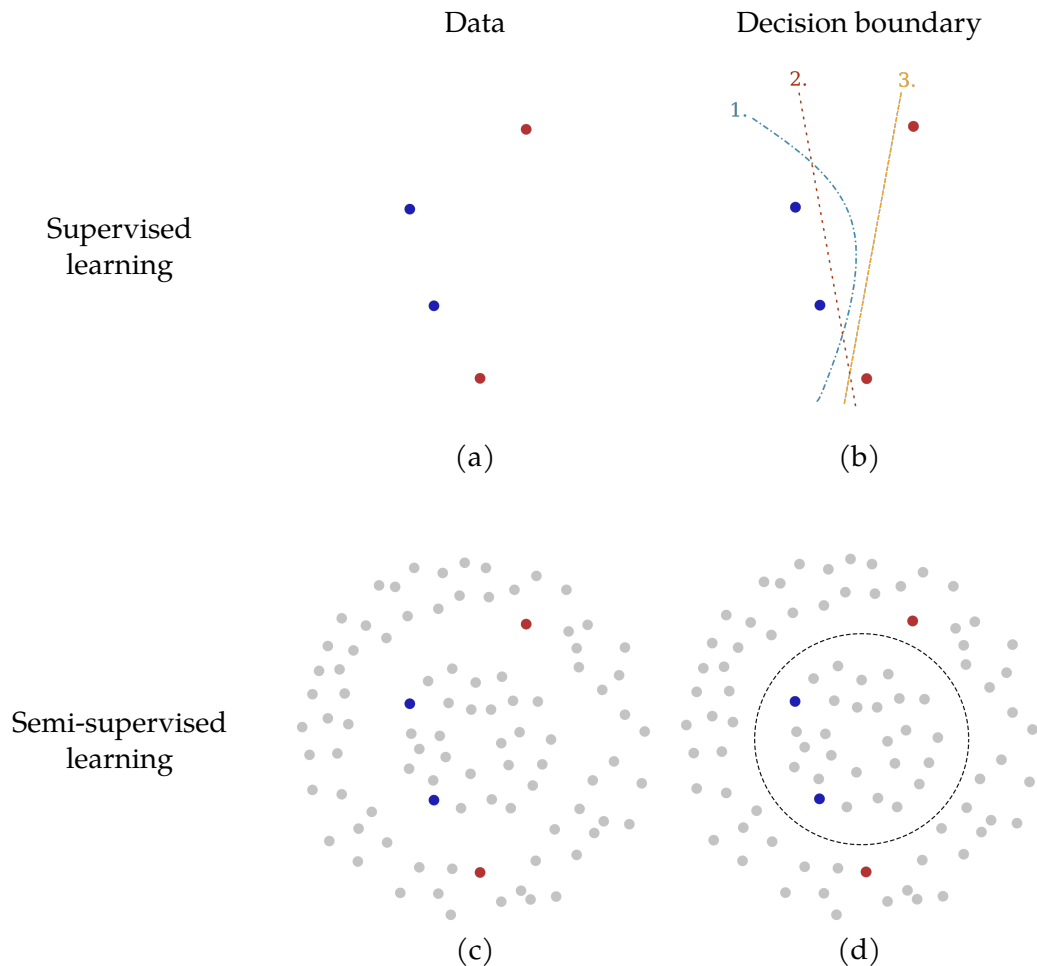


Figure 1.4 – Supervised vs. semi-supervised learning. Top: a supervised setting with very few labeled data is shown in (a), then how can an algorithm decide which is the best decision boundary? As illustrated by (b) 1, 2 or 3, seem all to be a reasonable choice. Bottom: a semi-supervised setting is represented. In (c) one has access to the same labeled data, but additional unlabeled data are available. Thanks to these new data, one can leverage information about the data distribution to determine a better suited decision boundary, like in (d).

1.2.4 Discriminative vs. generative models

Traditionally, there are two kinds of algorithms for supervised learning: discriminative models and generative models.

Discriminative models estimate the conditional distribution $p(y|x)$ of targets, given inputs. In other words, they try to find a map f from the input space to the output space such that for each (x_i, y_i) pair in the training data, $f(x_i) = y_i$. The main goal of discriminative models is to find the decision boundaries between categories.

Generative algorithms instead try to model the joint density $p(x, y)$ of inputs and labels, the idea is to understand how data pairs $\{(x_i, y_i)\}_{i=1}^n$ have been generated. Then a predictive density $p(y|x)$ can be derived by the means of the Bayes theorem:

$$p(y|x) = \frac{p(x, y)}{p(x)}. \quad (1.1)$$

Generative models can also be used for unsupervised learning when they model the distribution of data (without label information), $p(x)$.

Figure 1.5 illustrates the conceptual differences between these two families of algorithms.

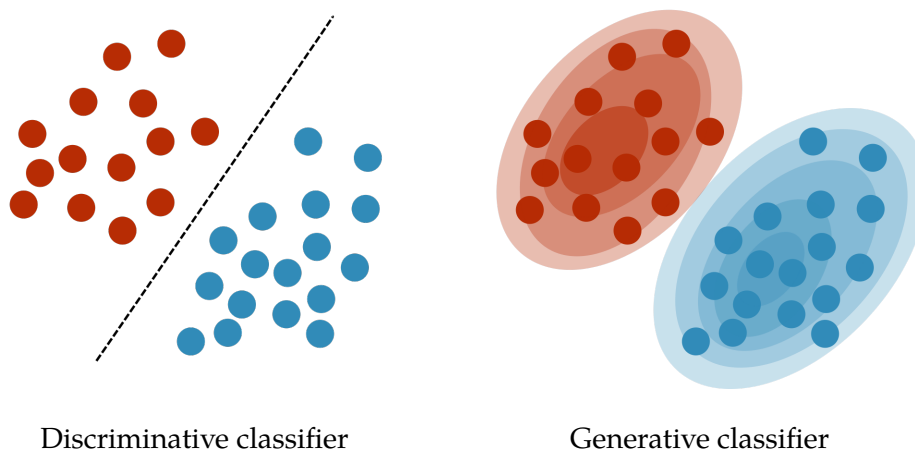


Figure 1.5 – Discriminative vs. generative classifier. A discriminative classifier predicts a decision boundary between classes (left). A generative classifier estimates class distributions (right).

1.3 Deep learning

Deep learning is a subfield of machine learning. One of the main issues of traditional machine learning algorithms is that they require manual feature preprocessing steps for inputs to be fed into the models; which made the performances of these algorithms extremely dependent of the quality of human-generated features.

In contrast, deep learning techniques have been created to learn representations directly from raw inputs, avoiding the feature engineering stage. This is achieved by introducing representations that are expressed in terms of simpler ones, by composing layers and layers of functions and non-linear terms, usually called *neural networks*.

The term “neural” refers to the origins of deep learning, since its basic components – artificial neurons – were inspired from neuroscience and brain functions. However, current deep learning research is guided by many mathematical and engineering disciplines, and does not represent a model of the brain.

In what follows, we present a brief history of deep learning and the main neural network architectures that contribute to the analysis of remote sensing imagery, and that serve as basis to this work.

1.3.1 Brief history of deep neural networks

Today’s deep learning approaches and success have been slowly built on the theoretical and practical contributions of several researchers over the last seven decades, from the first artificial neuron model, through backpropagation methods, to the computational power and training data [30].

In 1943, McCulloch and Pitts [31] proposed a first artificial neuron to model brain function. It was just a linear model that could recognize two different classes. It had no learning mechanism and weights needed to be set by the user. In 1950, Rosenblatt [32] presented the perceptron, the same neuron model with learning capabilities to perform binary classification. This was the first model that could learn weights from samples. These two contributions settled the basis of deep neural networks as we know them.

In 1960, Kelley [33] derived the first version of the backpropagation algorithm that is widely used today to train modern neural networks. Two years later, Dreyfus [34] developed a backpropagation algorithm based on the derivative chain rule. These two works, even if not related to deep learning yet, were the first steps toward backpropagation.

The “winter” of neural network research came in 1969, when Minsky and Papert [35] demonstrated that the perceptron was not able to solve complicated functions like XOR. Their work showed the limitations of linear models.

In 1970, Linnainmaa [36] published a general method for automatic differentiation, that efficiently computed the derivative of a differentiable composite function that could be represented as a graph, by recursively applying the chain rule to the building blocks of the function. He also implemented it as computer code. However, backpropagation was not used for neural network training till the next decade.

Between 1965 and 1971, Grigoryevich worked on deep neural networks, training them using group method of data handling [37], because of these contributions he is sometimes called the *father of deep learning*.

The Neocognitron [38] of Fukushima in 1980 can be considered as the first convolutional neural network architecture, a mechanism able to recognize visual patterns such as handwritten characters.

The 70’s and 80’s were the times where the link between backpropagation and neural networks was made. Indeed, Werbos proposed in his PhD thesis [39] the use of this mechanism of differential programming for neural networks’ training. In the 80’s, Rumelhart [40] showed the successful implementation of backpropagation for training neural networks, while LeCun et al. [41] proposed a multi-layer architecture with a first convolutional layer, trained by backpropagation, to recognize handwritten digits². These contributions led to the wide practical adoption of backpropagation to train deep learning methods in the future, they opened the gates for training complex deep architectures, which was the main issue in the early days of neural networks.

In terms of theoretical advances on learning with deep neural networks, Cybenko [42] in 1989 presented the proof for the *Universal approximation theorem*, that establishes that feed-forward neural networks with one hidden layer containing a finite number of neurons can approximate any continuous function. It further adds credibility to Deep Learning.

Other important contributions over this time, that we will not detail in this document, but deserve being mentioned are: Hopfield networks (1982), Boltzmann machines (1985), restricted Boltzmann machines (1986), the first LSTM, deep belief networks, etc.

However, even if all these contributions settled the bases of deep learning as we

2. https://www.youtube.com/watch?v=FwFduRA_L6Q

know it today, they were not enough to show the full potential of neural networks. The community, at the time, lacked of the last two ingredients: computational capacities and large training databases.

In 2005 and 2006, the first deep networks trained on GPUs appeared [43, 44]. From 2008, a research group from Stanford started advocating for the use of GPUs for training Deep Neural Networks to speed up the training time by many folds. This brought practicality in the field of Deep Learning for training on huge volume of data efficiently.

Finding large labeled databases has always been an issue for the deep learning community. It was in 2009 that Fei-Fei Li et al. published the ImageNet dataset [45], with the ambition to expand and improve the available data to train AI algorithms. ImageNet was the first large-scale labeled image database, containing 3.2 million images in total. At present, ImageNet counts more than 14 million hand-annotated images and has greatly contributed to computer vision research. In 2010, the first ever ImageNet Large Scale Visual Recognition Challenge (ILSVRC) was organized. Researchers compete to correctly classify and detect objects and scenes, every year ever since.

2012 defined a turning point in the history of deep learning, when Krizhevsky et al. won the ImageNet classification challenge with AlexNet [46], a GPU implemented convolutional neural network. AlexNet achieved a top-5 error of 15.3%, with more than 10% of margin over the closest competitor. This event triggered a new deep learning boom globally.

The relevance of deep learning research nowadays has been confirmed in 2019, when Yoshua Bengio, Geoffrey Hinton and Yann Lecun were distinguished with the Turing award 2018 for their contributions in the field of deep learning and artificial intelligence [47].

1.3.2 The multilayer perceptron

Artificial neurons are the cornerstones of deep learning, they are the basic unit of deep neural networks.

The perceptron proposed by Rosenblatt in 1957 was biologically inspired by our brain, and the structure of our neurons. Fig. 1.6 illustrates the similarities between an artificial neuron and a biological neuron.

However, the perceptron conceived by Rosenblatt is not exactly the same as the one presented in Fig. 1.6. Indeed, Rosenblatt's perceptron was just a linear model, without the activation function at the end. Deep composition of these non-linear functions is an

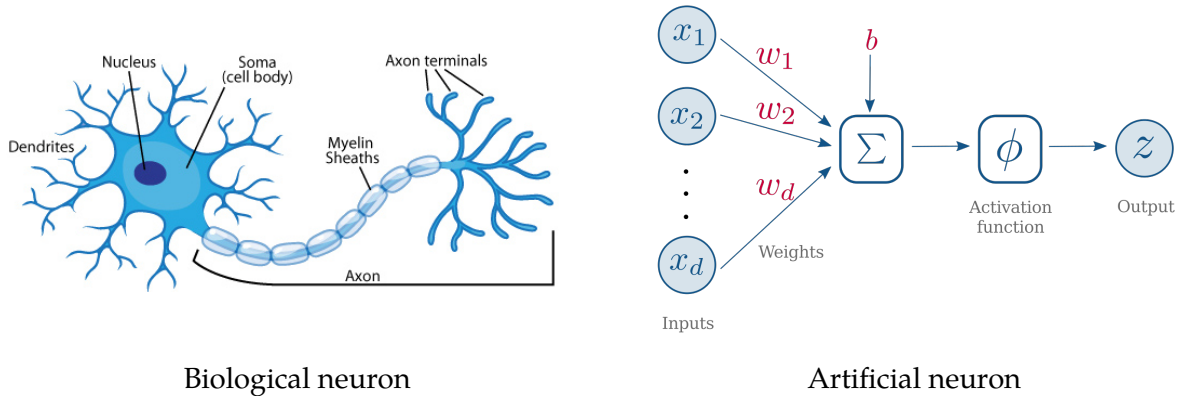


Figure 1.6 – Comparison between a biological neuron and an artificial neuron. At the beginning, artificial neural networks were inspired from the mechanisms of biological neurons. Today, they are mathematical models, functions approximating machines, that do not intend to depict brain functions. Figure partially taken from [ASU school of life sciences](#).

essential element for the success of deep learning to learn representations.

An artificial neuron, as we know it today, is defined by a set of parameters $\theta = \{\mathbf{w}, b\}$ and a non-linear activation function $\phi : \mathbb{R} \rightarrow \mathbb{R}$. Vector $\mathbf{w} \in \mathbb{R}^d$ is usually known as the *weights* and $b \in \mathbb{R}$ is known as the *bias*. The goal is to produce an output $z \in \mathbb{R}$ from an input vector $\mathbf{x} \in \mathbb{R}^d$, by the means of the parameters θ , and the function ϕ as described in equation

$$z = \phi(\mathbf{w}^\top \mathbf{x} + b). \quad (1.2)$$

The simplest neural network is the single layer perceptron (SLP)³, which is an extension of the artificial neuron previously defined. In a SLP, the input x passes through a set of m neurons in a parallel way, to compute different activations (with different parameters) for the same input. This can be expressed as:

$$\mathbf{z} = \phi(\mathbf{W}^\top \mathbf{x} + \mathbf{b}), \quad (1.3)$$

where $\mathbf{W} \in \mathbb{R}^{d \times m}$ is a matrix of weights and $\mathbf{b} \in \mathbb{R}^m$ is a bias vector. The activation function ϕ performs element-wise operations. Fig. 1.7 (left) shows a representation of a single layer perceptron.

Multilayer perceptrons (MLP) consist in a composition of several SLP, that make

3. also known as *fully connected layer*.

the multiple *layers* of these architectures:

$$y = f(\mathbf{x}; \boldsymbol{\theta}) := f^{(k)}(f^{(k-1)}(\dots (f^{(1)}(\mathbf{x}; \boldsymbol{\theta}^{(1)})) \dots; \boldsymbol{\theta}^{(k-1)}); \boldsymbol{\theta}^{(k)}). \quad (1.4)$$

MLP are also known as **feedforward neural networks** because information flows through the function being evaluated on \mathbf{x} , through the intermediate computations $f^{(i)}$ used to define f , to finally yield the output y , with no feedback connections. They can be associated with a directed acyclic graph describing how the functions are composed together, as shown in Fig. 1.7.

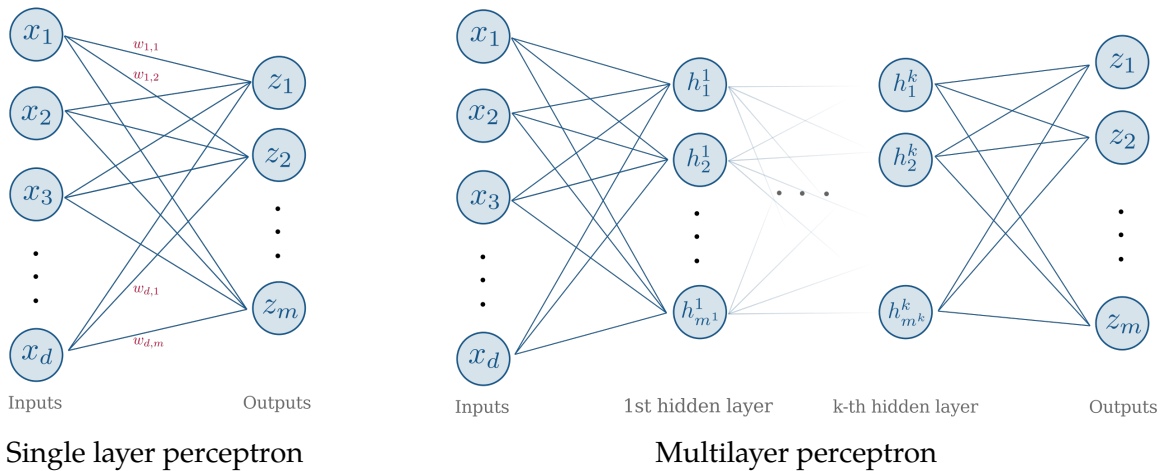


Figure 1.7 – Left: Single layer perceptron (SLP) representation, with m artificial neurons (as in Fig. 1.6). Right: Multilayer Perceptron (MLP) with k hidden layers, each hidden layer i consists of m^i artificial neurons.

The choice of the activation function ϕ has been extensively studied over time. Indeed, it can have a direct impact on the convergence of the learning algorithm or its performance. For instance, ReLU [48] was presented as a solution to the vanishing gradient problem [49, 50], observed when training neural networks. Common activations functions are: the sigmoid function, hyperbolic tangent, rectified linear unit (ReLU), leaky rectified linear unit (LeakyReLU), to name a few. For a more detailed description about activation functions, the reader can see [51].

The goal of neural networks is to approximate a function f^* , such that $y = f^*(\mathbf{x})$ maps an input \mathbf{x} to an output y . To this end, we define a function $f(\mathbf{x}; \boldsymbol{\theta})$ as in Equation (1.4) and optimize the set of parameters $\boldsymbol{\theta}$ to obtain a good approximation $f(\cdot; \boldsymbol{\theta}) \approx f^*(\cdot)$.

To optimize the set of parameters of a neural network, one needs to define an objective function to minimize (or maximize). In the context of deep learning, we refer to this objective as the *loss function*, represented by \mathcal{L} . The loss function is usually a measure of the approximation error of the neural network over a set of training data \mathcal{D} . Training a neural network means solving the following minimization problem:

$$\hat{\theta} = \arg \min_{\theta} \mathcal{L}(f_{\theta}, \mathcal{D}). \quad (1.5)$$

The choice of the loss function to use depends on the task that we intend to solve. In the case of classification and segmentation problems, a standard choice is the *cross-entropy loss*.

A consequence of the introduction of non-linearities into neural networks is that the loss function to optimize is –usually– non-convex. Thus, there is no guarantee that a global solution to the optimization problem in Equation (1.5) exists. Moreover, even if global minima existed, it would be prohibitively costly to find them. For this reason, neural networks are usually trained by using iterative, gradient-based estimators, combining backpropagation with stochastic gradient descent algorithms.

1.3.3 Convolutional neural networks

Convolutional neural networks (CNNs) have been at the center of significant advances in deep learning, especially in computer vision. Even though CNNs have been used in the 90's to recognize handwritten characters [41, 52], it is since 2012 –with AlexNet's [46] victory in the ImageNet classification challenge– that CNNs have become the undisputed method for deep image processing.

Feedforward networks have been successful in many applications, however –because of their fully connected layers– they consider all elements of input x equally, disregarding all spatial information that might be available (for instance, if x represents an image). In view of the above, CNNs have been developed as specialized architectures for processing data that have a known, grid-like topology. Image data, in particular, can be thought of as a 2D grid of pixels.

CNNs are neural networks that use convolution⁴ in place of general matrix multipli-

4. Convolution is a mathematical operator. Let f and g be two functions, the convolutional product $f * g$ is defined by: $(f * g)(t) := \int_{-\infty}^{+\infty} f(\tau)g(t - \tau)d\tau$.

cation in at least one of their layers. Usual CNN architectures have three components: convolutional layers (including the non-linear activation function), pooling layers and fully connected layers. Fig. 1.8 shows the standard architecture of a CNN.

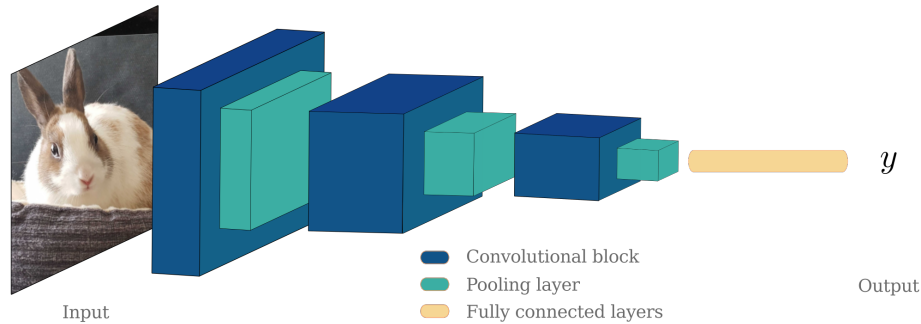


Figure 1.8 – Standard CNN architecture. A CNN is usually composed of convolutional blocks (sequence of convolutional layers, plus activation function, pooling layers and fully connected layers at the end).

Convolution. Since we are interested in CNNs for image processing, in this section we describe 2D convolutions (although they can be defined for any dimension).

Let $I \in \mathbb{R}^{W \times H \times C}$ represent an image of size $W \times H$ with C the number of channels, and $\mathbf{K} \in \mathbb{R}$ a convolutional kernel the basic 2D convolution⁵ is defined as:

$$\mathbf{K} * \mathbf{I}[m, n] = \sum_{c=1}^C \sum_{i=-p}^p \sum_{j=-q}^q \mathbf{I}[m+i, n+j, c] \cdot \mathbf{K}[i, j, c]. \quad (1.6)$$

To avoid border issues because pixel values are not defined outside an image, as well as to keep spatial dimensions of the images after convolutions, *padding* methods can be applied.

A **convolutional layer** is the result of parallel convolutional kernels applied to the same input image (in an analogous way to the MLP described above, Section 1.3.2), combined with a bias term and activation function.

The convolution operation leverages three important ideas that can help improve a machine learning system: sparse interactions, since the convolutional kernel is smaller

5. Strictly speaking, this is not a convolution. For practical reasons, we use the cross-correlation operator instead.

than the input; parameter sharing, since the same kernel is applied all over the input; and equivariant representations with respect to translations. Moreover, convolution provides a means for working with inputs of variable size. For a complete guide to existing convolution operators, we refer the reader to [53].

Pooling. A pooling function transforms feature maps (the output of a certain layer of the network) by applying a summary statistic of nearby features. They provide an approach to reduce the size of representations (operation known as *downsampling*), which reduces the memory costs. Pooling layers also provide invariance to small translations of the inputs. Common pooling methods include *max pooling* or *average pooling* that report the maximum and the average of a rectangular neighborhood, respectively.

1.3.4 Fully convolutional networks

Convolutional neural networks revolutionized the way we performed image classification and other tasks with structured outputs (like object detection). However, the size reduction done by pooling layers and the fact that fully convolutional layers break all the spatial information prevent CNNs to make dense, pixel-wise predictions, which are the expected outputs for semantic segmentation.

To solve this issue, Long et al. [23] proposed the first *fully convolutional network* (FCN). The key idea of this architecture is to replace the fully connected layers at the end of a CNN, by convolutional layers. Making this simple modification enables the preservation of spatial information. Then, feature maps are upsampled to the original input size to obtain pixel-wise predictions. Fig. 1.9 illustrates this architecture.

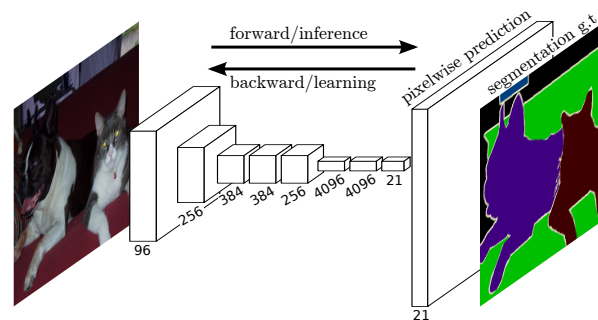


Figure 1.9 – Fully convolutional neural network. Figure from [23].

This work represented a turning point in image segmentation. It demonstrated that neural networks can be trained end-to-end on variable-sized images to obtain dense

predictions. Many improvements to this basic FCN architecture have been presented in the literature over the years to get better semantic segmentation results, and current state-of-the-art methods for this task still inherit from it [21].

Standard semantic segmentation networks are based on an encoder-decoder architecture, inspired from convolutional autoencoders [54]. The encoder part is usually based on CNN architectures (without fully convolutional layers) and the decoder consists in progressive upsampling steps that mirror the encoder operations. This enables gradual recovery of spatial features during the decoding process.

Popular networks that follow this approach are SegNet [55] that consists of an encoder based on a VGG-16 architecture, followed by a symmetric decoder and a pixel-wise classification layer. Its main novelty is the upsampling method: it uses pooling indices computed in the max-pooling step of the corresponding encoder layer to perform non-linear upsampling. This removes the need for learning to upsample [56]; meanwhile, U-Net [57] incorporates *skip-connections* to copy and concatenate the encoder's feature maps to the input of the corresponding decoder's layer. Unlike SegNet, upsampling is achieved via transposed convolutions.

Over the years, new architectures have been proposed –based on this encoder-decoder principle– with new characteristics that improve semantic segmentation results. Some improvements include changing the encoder architecture for a more efficient CNN (like ResNets [58]), using different kinds of convolution operators or pooling layers, or adding post-processing steps to refine the outputs. Among the main contributions we find Deeplab [59], LinkNet [60] and PSPNet [61].

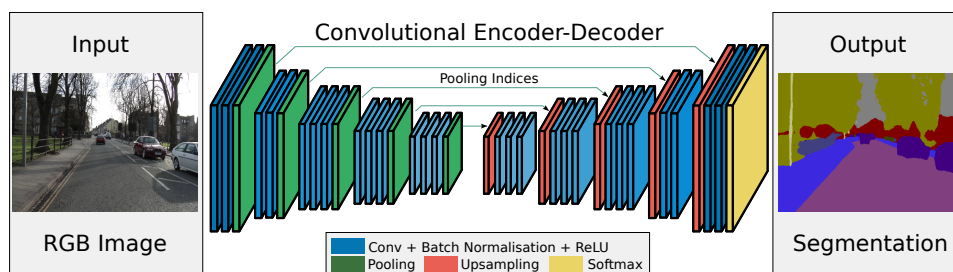


Figure 1.10 – SegNet. Figure from [55].

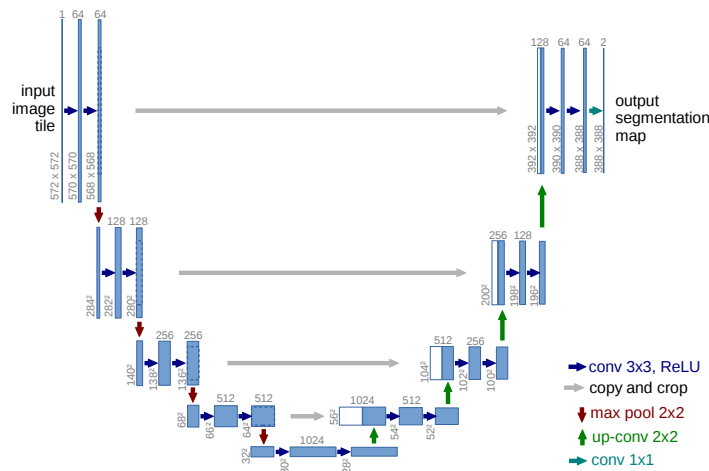


Figure 1.11 – U-Net. Figure from [57].

1.3.5 Deep semi-supervised learning

Deep learning has shown remarkable performances on a wide range of *supervised tasks*, when trained on large amounts of labeled data. However, collecting large, annotated datasets requires considerable efforts, resources and time; which is infeasible in many practical applications. On the other hand, nowadays we live the “big data era”, and unlabeled samples are –in most cases– easily available. Therefore, interest in **deep semi-supervised learning** techniques has been rising, trying to move forward and develop less label-dependent deep learning approaches.

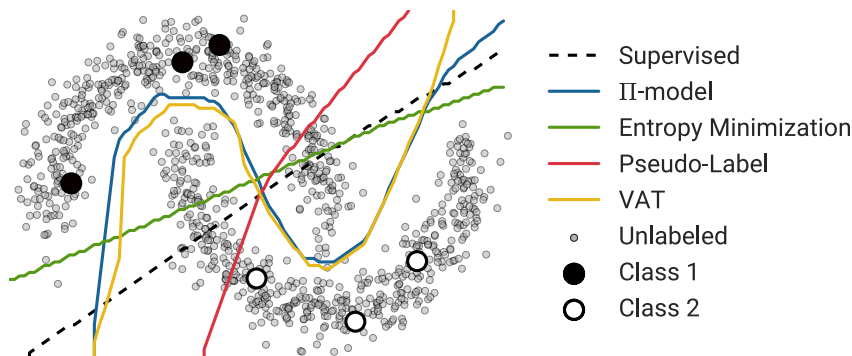


Figure 1.12 – Deep semi-supervised learning. Figure from [62].

Broadly speaking, semi-supervised methods in deep learning can be divided into the following main categories⁶:

6. Another class of methods considers that data are represented as the nodes of a graph where la-

- (i) **Self-training**⁷. A supervised model is trained on labeled data. An iterative process is then applied to an extended labeled training set, with self-generated labels. More details in Chapter 3.
- (ii) **Consistency training**. Models of this kind work under the assumption that slightly modified versions of the same input should have semantically similar outputs. More details in Chapter 3.
- (iii) **Generative models**. These models leverage unlabeled samples to estimate the data distribution $p(x)$, in the hope to learn significant features to better estimate $p(y|x)$. More about generative approaches in Chapter 4.

In this context, several works have emerged. In what follows we mention some of the main contributions, but it is not an extensive review of all existing methods. For a complete overview of semi-supervised methods in deep learning, the reader can see [63]. The Π -model [64] was one of the first works to exploit consistency training, proposing a consistency loss term between different stochastic outputs of a neural network f_θ for the same input. Mean teacher [65] refines the Π -model by setting the target predictions as the exponential moving average of parameters from previous training steps. Instead of relying on the stochasticity of f_θ , virtual adversarial training [66] proposes to modify directly the input x through a perturbation r_{adv} and apply a consistency regularization term. Pseudo-labeling [67] produces “pseudo-labels” for unlabeled data using the prediction function itself over the course of training. Pseudo-labels are retained according to a certain rule (e.g. a threshold over the confidence of the prediction). Fig. 1.12 shows how semi-supervised methods work on a toy example.

Generative models have also been developed to tackle the semi-supervised classification, including semi-supervised VAEs [68] and semi-supervised GANs [69, 70].

Lately, holistic methods combining some of the precedent ideas have emerged, setting new state-of-the-art results on semi-supervised classification: MixMatch [71], ReMixMatch [72], FixMatch [73].

Semi-supervised methods for semantic segmentation in deep learning have been developed in the last years, but mostly in the form of weakly supervision: from scribbles [74, 75], bounding boxes [76, 77], and image-level annotations [77] to obtain dense, pixel-wise predictions. Pseudo-labels have also been used to address the semi-supervised

bel information is meant to be propagated from labeled nodes to unlabeled ones. These graph-based methods will not be studied in this thesis.

7. also known as *proxy-label methods* or *pseudo-labeling*

semantic segmentation problem [78], propagating labels from annotated samples through non-annotated ones, according to a confidence criterion, to artificially enlarge available training data. Other works include unlabeled data during training in a generative adversarial network framework [79, 80]. The method in [81] proposes a multi-task method (comparable to the approach we develop in Chapter 3), but the final goal is to achieve domain adaptation.

More recently, consistency regularization-based approaches have been extended to semantic segmentation [82–84].

Finally, a generative approach based on GANs to learn the joint distribution of image and semantic map pairs, $p(\mathbf{X}, \mathbf{Y})$, has shown impressive results in semantic segmentation and out-of-domain generalization [85]. Nevertheless, as pointed out by its authors, this approach is limited to simple data following a unimodal distribution and cannot address complex images.

1.3.6 Deep learning in Earth observation

Previous sections have presented deep learning methods that transformed the way the computer vision community tackles tasks such as image classification, object detection or semantic segmentation on natural, everyday images.

Even though remote sensing data share some similarities with natural images –they are both structured data–, EO imagery uses different types of sensors and very specific points of view. Therefore, computer vision and deep learning techniques should not be transposed recklessly to remote sensing data processing. Instead, we should consciously take into account the particularities of these data and combine computer vision and deep learning methods with the existing knowledge of the remote sensing community, making the best of both worlds.

Indeed, remote sensing data come with new challenges for deep learning, since satellite and aerial imagery comes in different forms, and contains very rich information about the Earth’s surface. Thus, remote sensing image analysis raises new questions about how to exploit the abundant information available on this kind of data [86]. Among particular characteristics of remote sensing data we can mention:

Multimodality. One can possess different data sources for the same area, such as optical sensors (multi or hyperspectral) and synthetic aperture radar (SAR).

Geolocalization. EO data are naturally located in the geographical space and this in-

formation is available.

Time series. The temporal component of EO data is increasingly relevant. Indeed, satellites acquire data of the entire planet periodically. This collection of periodic images defines time series that can (and should) be exploited⁸.

Big data. Since satellites image the Earth in a short period, data volumes grow extremely fast and at a global scale.

In the last decade, many efforts have been made and the use of deep learning techniques for the analysis, interpretation and processing of remote sensing data has grown exponentially [86, 87]. Indeed, several EO tasks are today tackled using deep learning techniques: change detection [88, 89], building detection [90], scene classification [91], building height estimation [92], data fusion [93].

In the last years, deep learning in Earth observation has gone through three main, overlapping phases [94]: (i) exploration, direct transfer of deep learning and computer vision techniques to EO applications; (ii) benchmarking, train and compare deep learning models on EO data, study generalization capacities and proposing large-scale datasets in the domain to encourage further research; and (iii) EO-driven methodological research, going beyond what we have achieved. Regarding this last point, Tuia et al. [95] propose some guidelines for future research in the field.

Since semantic segmentation is at the heart of this work, we devote a small section to give more insight on the progress of semantic segmentation techniques in Earth observation.

Deep semantic segmentation of EO data. Before the deep learning era, the remote sensing community was already aware of the importance of texture information to obtain good semantic segmentation maps. Therefore, filters based on convolutional operators were used as preprocessing steps to extract features that would be fed to more traditional classification methods, like random forests or support vector machines [94].

Therefore, moving forward to the use of convolutional neural networks and fully convolutional networks for semantic segmentation was a reasonable transition. In 2015, the first works using CNNs to achieve semantic segmentation appeared. Lagrange et al. [96] applied CNNs over image patches (defined by sliding windows) and assigned a label to the central pixel of each patch; and showed that this deep-learning-based

8. While the methods proposed in this thesis do not specifically address satellite image time series, they could be extended to such data.

approach was superior to other traditional machine learning methods. Fully convolutional methods quickly replaced this patch-based approach, allowing us to accelerate inference times and making better use of spatial information. Sherrah et al. [97] first explored FCN to achieve dense semantic labeling on aerial data. From then on, semantic segmentation has mostly been performed using encoder-decoder architectures [98–102].

However, training deep neural networks for semantic segmentation in EO has only been possible thanks to the great effort of the community to provide public datasets. We will delve into these contributions in the next chapter (Chapter 2).

Semi-supervised learning in Earth observation. Semi-supervised learning methods are especially appealing for the remote sensing community, since EO data are naturally well-suited in this context. Indeed, labeled data are hard to obtain, while raw (unlabeled) data are constantly gathered through satellite or aerial missions. Thus, semi-supervised methods are a feasible solution to improve the classification performances and the generalization capacities of our models.

In the last decades, several semi-supervised methods have been proposed for Earth observation data applications. Before the deep learning outbreak, different approaches have been explored, including graph-based methods to integrate unlabeled data into the learning process [103, 104]; use unlabeled examples to achieve manifold alignment of data coming from different modalities [105]; and factor analysis for hyperspectral image classification [106]. More recently, deep semi-supervised learning techniques have emerged, but most of them rely on self-training and pseudo-labeling [107]: combining them with other techniques to build more robust models such as the use of an ensemble of CNNs to assign pseudo-labels and prevent error propagation [108]; using cross-modal data [109]; applying sample selection schemes to train transferable deep models for land use classification [110, 111]; or using stacked auto-encoders and soft-label propagation to tackle the building detection problem [112].

Other strategies that use semi-supervised learning in remote sensing applications include: a center-based discriminative adversarial learning framework for cross-domain land cover classification of aerial images [113]; integrating CNNs and active learning to better use unlabeled samples for hyperspectral image classification [114]; the use of a semi-supervised shallow network, self-organizing map framework, to classify and estimate physical parameters from multispectral and hyperspectral images [115]; and

using multi-attention and an adaptive kernel for semi-supervised classification of multispectral images [116].

Fewer are the works that exploit generative models to leverage unlabeled samples for training. GANs have been used to extract features from hyperspectral images for semi-supervised classification [117], or jointly with gated attention and a discriminative network for scene classification of aerial images [118]. A modified GAN, with a classifier as discriminator, has been developed to tackle the multispectral scene classification problem [119].

Throughout this manuscript, we study semi-supervised learning for large-scale EO data understanding and semantic mapping. We tackle the problem from different perspectives: first, we analyze the data, study existing datasets in EO and supervised learning techniques and their limitations; secondly, we investigate discriminative approaches based on multi-task learning and consistency training; and finally, we examine generative approaches for semi-supervised scene classification in EO.

THE POTENTIAL OF SEMI-SUPERVISED LEARNING IN EARTH OBSERVATION

Contents

Chapter summary	58
2.1 Current Earth observation benchmarks	59
2.2 The necessity of new training paradigms and large-scale EO datasets	62
2.2.1 Analysis of supervised learning on small-scale datasets	63
2.2.2 Supervised learning at large-scale	65
2.3 The MiniFrance suite	70
2.3.1 MiniFrance	70
2.3.2 TinyMiniFrance	75
2.4 Statistical analysis of the representativeness of training and test datasets	76
2.4.1 Appearance analysis	77
2.4.2 Class representativeness analysis.	81
2.5 Defining the labeled, unlabeled and test splits for MiniFrance	85
2.6 Comparing MiniFrance to classic datasets	87
2.7 Data fusion contest 2022: MF-DFC22	88
2.8 Conclusions	92

Chapter summary

The availability of large public vision datasets has been crucial for the considerable progress in computer vision that we have witnessed in the last decade. Indeed, they not only represent large amounts of training data, but also provide the means to compare the performance of competing algorithms.

This chapter presents an analysis of existing Earth observation datasets (see Section 2.1): are they representative of the real-life remote sensing use-cases? We study what are the applications of interest for the remote sensing community and, therefore, what are the desirable features of a trustworthy evaluation benchmark.

We also perform a critical analysis of current supervised approaches (cf. Section 2.2). More precisely, we investigate the learning capacities of supervised semantic segmentation networks on different settings: on small-scale datasets, and at a large-scale multi-location set-up. We observe that common supervised semantic segmentation techniques have generalization issues in the large-scale setting, when labeled data are not varied enough. Hence, new learning paradigms should be studied in the future, in order to develop methods that are well-suited for real-life Earth observation applications.

In view of the above, in Section 2.3 we present the **MiniFrance benchmark**, a novel large-scale dataset, especially designed for semi-supervised semantic segmentation. MiniFrance has several unprecedented properties: it is large-scale, containing over 2000 very high resolution aerial images (at a sub-meter resolution); it is varied, covering 16 conurbations in France, with various climates, different landscapes, and urban as well as countryside scenes; and it is challenging, considering land use classes with high-level semantics. Nevertheless, the most distinctive quality of MiniFrance is being the only dataset in the field especially designed for semi-supervised learning: it contains labeled and unlabeled images in its training partition, which reproduces a life-like scenario. Along with this dataset, in Section 2.4 we present tools for data representativeness analysis in terms of appearance similarity and a thorough study of MiniFrance data, demonstrating that it is suitable for learning and allows us to measure the generalization capacities of algorithms in a semi-supervised setting.

2.1 Current Earth observation benchmarks

The tremendous progress of computer vision –where machine learning is applied on images– in the last decades would not have been possible without the development of large public datasets, such as ImageNet [45], COCO [120] or Cityscapes [121] for learning on visual data. These datasets do not only supply a source of large amounts of training data, but also provide the means to fairly compare learning algorithms. They allow us to test their scalability and reliability. The availability of these public benchmarks is the key to improve the performance of our models, to explore their strengths and weaknesses and thus, push the research limits further.

In the same vein, the remote sensing community has also published several datasets for different tasks in order to encourage the research in the field [122]. Indeed, the classical IEEE GRSS Data Fusion Contest¹ (DFC) has been running every year from 2006 [123]. Moreover, several works in the last decade have shown that remote sensing data analysis can truly benefit from the automatized treatment of images by using deep learning approaches [2, 94]. Combining this kind of methods with domain knowledge could greatly accelerate remote sensing image processing, allowing for real-time monitoring of the Earth and our environment. Table 2.1 describes some of the main initiatives to develop Earth observation datasets for various tasks of interest².

As stated by Torralba et al. [143], although the availability of public datasets has been responsible for much of the recent progress in computer vision, there are still major issues that the research community needs to keep in mind. For instance, one of the main problems of having static, unchanging benchmarks is the intrinsic gradual overfitting, since algorithms become too adapted to the dataset over time. This leads to another important matter: the lost of focus of the community on the real objective, as much of the research works concentrate on gaining a few accuracy points over one benchmark, making mostly incremental contributions to the field. The most fundamental question is, though, *are our datasets measuring the right thing?*, are they measuring the expected performance of models in a real-world task? In the quest of making our datasets a trustworthy *representation* of our world, we need them to integrate the richness and variabilities of real-life settings, avoiding biases. What we *need* is datasets that are able to measure the generalization capacities of our algorithms, according to the

1. <https://www.grss-ieee.org/technical-committees/image-analysis-and-data-fusion/?tab=past-data-fusion-contests>

2. However it is far from being an extensive list of all the existing remote sensing datasets.

Table 2.1 – Earth Observation datasets summary. *

<i>Vision Task</i>	<i>Dataset</i>	<i>EO task</i>	<i>Location</i>	<i>Zone type</i>	<i>Surface (km²)</i>	<i>Resolution (cm/px)</i>	<i>Number classes</i>
<i>Classification</i>	EuroSAT [124]	LC, LU	Europe	Urb., Ctry	11,000	1,000	10
	So2Sat LCZ42 [125]	LC, LU	Worldwide	Urb.	~ 51,000	1,000	17
	AID [126]	LC, LU	Worldwide	Urb.	-	Variable	30
	UCMerced [127]	LC, LU	USA (various regions)	Urb.	~ 12	30	21
	BigEarthNet [91]	Multi-label LC, LU	Europe (10 countries)	Urb., Ctry	850,000	1,000	~ 40
	DENOTHOR [128]	Crop monitoring	Northern Germany	Ctry	1152	300	9
<i>Object detection</i>	DOTA [129]	OD	Worldwide	Urb.	-	Variable	15
	xView [130]	OD	Worldwide	Urb.	1415	30	60
	xBD [131]	CD, Build.	15 countries	Urb.	45,362	30	-
<i>Semantic Segmentation</i>	Vaihingen [132, 133]	LC	Vaihingen (Germany)	Urb.	1	9	6
	Potsdam [132, 133]	LC	Potsdam (Germany)	Urb.	3.5	5	6
	Inria [134]	Build.	USA, Austria (10 cities)	Urb.	810	10 - 30	2
	DeepGlobe [135]	Road, Build., LC	Worldwide	Urb., Ctry	2,220/ 984/ 1,717	50/ 31/ 50	2/ 2/ 7
	Christchurch [136, 137]	LC, OD	Christchurch (New Zealand)	Urban	5	10	4
	HRSCD [138]	CD	France (2 areas)	Urb., Ctry	14,550	50	25
	SEN12MS [139]	LC	Worldwide	Urb., Ctry	~ 1.18 × 10 ⁶	1000	33
MiniFrance [J2]	LC, LU	France (16 areas)	Urb., Ctry	53,000	50	12	
<i>VQA</i>	RSVQAxBEN [141]	RSVQA [142]	Europe (10 countries)	Urb., Ctry	850,000	1,000	25 questions/patch

* Abbreviations: LC = Land Cover; LU = Land Use; Road = Road Extraction; Build. = Building Extraction; OD = Object Detection; CD = Change Detection; Urb. = Urban; Ctry = Countryside; VQA = Visual Question Answering; RSVQA = Remote Sensing VQA.

situations we want to model.

To illustrate the above questions, imagine that there has been an earthquake in the north of Chile. We would like to assess the damages very quickly by the means of an automatic algorithm. Even though we do not have data from the affected zone at our disposal, we have annotated data from a previous earthquake occurred in Christchurch, New Zealand. Would we be able to train a deep learning model able to assess the damages in Chile, being trained on such a different geographical zone as this New Zealand's city?

Indeed, in real-life Earth observation applications, one typically has access to annotated data from a specific geographic zone and/or from a specific sensor, and would like to apply a model to new data that could come from a different sensor or different geographic zone, with different climate, different season or different resolution.

One way to obtain models that generalize to unknown locations is to learn non-location-specific features. This can be achieved by training on several, diverse sites. If some of the Earth observation datasets mentioned in Table 2.1 already take into account multiple locations, most are limited to urban scenes only and/or they are devoted to a single class (such as buildings or roads) or to land cover (and not land use) classes. Land cover refers to the ground surface coverage: vegetation, urban infrastructure, water, etc; while land use indicates the purpose the land serves: urban, industrial buildings, agriculture, etc. The second has more socio-economic impact, because it provides further information about human activity in a given area, however extracting this information from images only remains a major challenge [144].

Furthermore, all the aforementioned EO datasets were designed for fully supervised learning, which does not correspond to the real practical³ case where huge amounts of imagery are available, but only a few images come with some labeled regions, from specific locations. Indeed, labeled data are usually limited –the labeling process requiring too much effort, time and expert knowledge– however, there are large amounts of unlabeled data available that are being generated continuously (e.g., Copernicus Sentinels can provide data of the entire Earth every 6 days) and that could be exploited by our learning algorithms. Therefore, we need datasets that mimic these conditions.

In the following section we perform experiments that demonstrate the requirement of large-scale, multi-location datasets in Earth observation, that correctly represent the challenges of real-life applications, together with new learning techniques, able to lever-

3. that we want to tackle

age unlabeled data during the training process to capture all the available information and characteristics of data.

2.2 The necessity of new training paradigms and large-scale EO datasets

As discussed in the [Introduction](#), EO data analysis contributes greatly to better understand our planet and its dynamics. Nowadays, EO data are easily available, thanks to initiatives like Copernicus or Landsat. However, data exploitation can still be a bottleneck, since it requires human interpreters, for example to identify tree species and study deforestation in a local ecosystem, or to find new buildings and measure growth of urban areas.

Deep learning methods have shown to be useful to address Earth observation problems. Indeed, many state-of-the-art algorithms for object detection and image segmentation or classification [93, 145] have been successfully applied to aerial and satellite images. They allow us to produce quickly and without human intervention precise semantic maps, in both urban and rural contexts. However, these learning algorithms rely heavily on the availability of large annotated image databases. And even if collaborative cartographic resources, like OpenStreetMap, can be used as annotations [134], these are restricted in terms of semantics (only roads, buildings, etc.) or geographic locations (being biased toward urban zones).

Therefore, the question of quantifying the influence of existing datasets on the models we learn arises. In addition, we aim to define what is required to make a good dataset for training EO data classification and segmentation algorithms that generalize well to new locations and that make a good representation of real-world problems.

In this section we perform an experimental analysis of the amount of data necessary to successfully achieve supervised learning. Moreover, we study the generalization capacities of current supervised approaches with respect to data variability. These experiments bring out a better understanding of the required data variability for a dataset to make a good representation of real-life challenges. They also reveal the weaknesses of standard supervised approaches in terms of generalization capacities, as they do not generalize properly to new geographic locations [146], as we will see in Section 2.2.2. In consequence, new learning strategies should be investigated; for instance,

one could consider leveraging unlabeled data during training through semi-supervised techniques.

2.2.1 Analysis of supervised learning on small-scale datasets

In this section, we aim to test the sensitivity of supervised learning to the amount of labeled data available for training on a small-scale dataset.

To this end, we perform experiments over the ISPRS Vaihingen dataset, since it represents a classical setting for evaluating models in the remote sensing community. Indeed, the ISPRS 2D Semantic Labeling datasets [133], Potsdam and Vaihingen, are probably the most widely-used datasets for evaluation of semantic segmentation in EO applications. Both datasets were proposed by the ISPRS Working Group III/4 “3D Scene Analysis” as a part of the 2D Semantic Labeling contest, providing –for the first time– standard benchmarks for evaluating object extraction methods in Earth observation.

ISPRS Vaihingen consists of 33 infrared-red-green tiles with a spatial resolution of 9cm/px and an average size of 2000px \times 1500px. Dense annotations are available on 16 tiles for 6 classes of interest: impervious surfaces, buildings, low vegetation, trees, cars and clutter. The associated benchmark being now closed, we perform experiments using 12 annotated tiles for training (*train*) and 4 tiles for evaluation (denoted by *val*).

The following experiments were designed to study the effect of the amount of labeled data available on the performance of supervised neural networks for semantic segmentation. In particular, we perform these experiments using SegNet [55], an encoder-decoder architecture that has already been successfully used on EO data in previous works [147, 148]. We gradually reduce the amount of annotated images used for training. In the case of Vaihingen, we reduce the available images from 12 tiles to only one, while *val* remains unchanged. We repeat the experiment four times to get more statistically significant curves. Results are presented in Fig. 2.1.

The outcomes of this experiment are somehow surprising. When reducing the number of training tiles from 12 to 1 (only 8% of original data!), we report a decrease of *only* 12% of overall accuracy (from 90% to 78%) and 21% of mIoU (from 77% to 56%), i.e. much less than one would expect. Indeed, we supposed that reducing the number of training tiles would seriously impact the performance of the network. One possible reason is that all the images in the Vaihingen dataset are alike, thus, to generalize on them is a relatively easy task. However, one can note that training with more data is nevertheless preferable in terms of reliability: the variance increases as the number of tiles

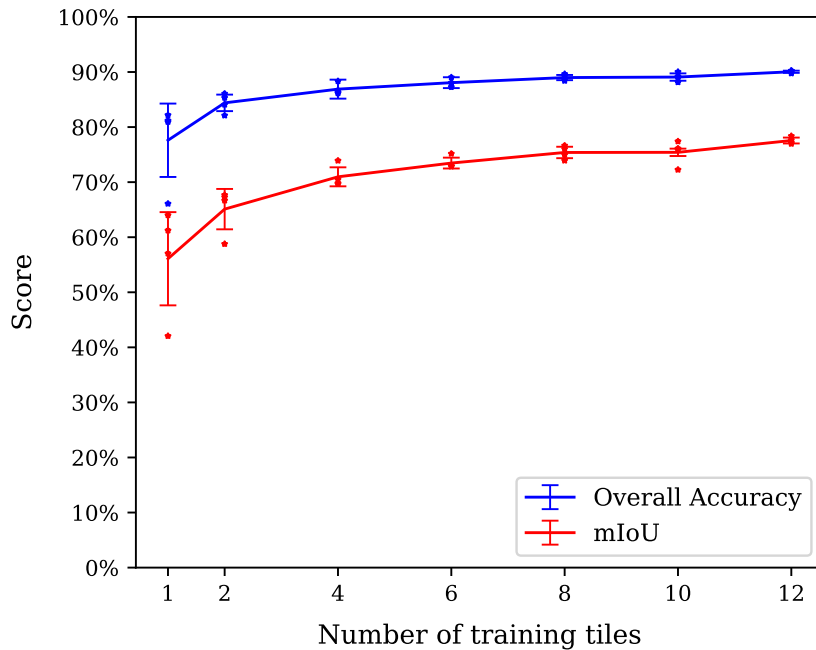


Figure 2.1 – Influence of the training set size (number of tiles) on the network performances, in terms of overall accuracy and mean Intersection over union (mIoU). The curves show the mean and the standard deviation for each score and * shows raw results.

decreases.

To better understand the quantitative scores from Fig. 2.1 in terms of segmentation quality, Fig. 2.2 shows the different predictions obtained for tile 30. We can observe that the quality of the segmentation map decreases notably when less annotated tiles are used during the training phase, with borders being less precise and shapes approximate. It is interesting to note that there is not a considerable difference between training with 10 tiles and with 6 tiles, however there is a greater difference when training with 1 tile: borders are less regular and little objects (such as cars) are not well learned, which explains why the mean IoU decreases faster on Fig. 2.1.

To assess the idea that the Vaihingen dataset has much redundancy, we observed its statistical distribution. In Fig. 2.3(a) and (b), we compare the color histograms over the 3 channels for *train* and *val*. Indeed, they are almost identical, which indicates that learning on a single location might not be so challenging. Actually, this is even promising in terms of practical business applications, since mapping one single area can be achieved after labeling only a few images. Nevertheless, it is not representative of the

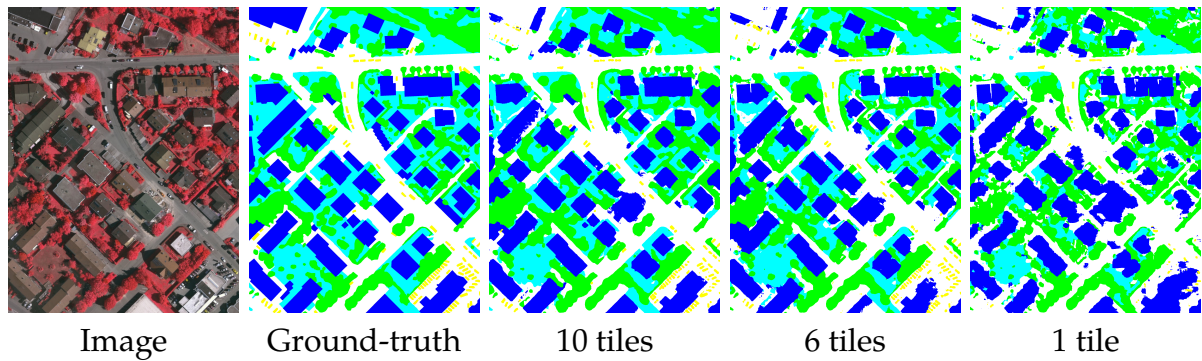


Figure 2.2 – Semantic maps obtained by reducing the available amount of labeled tiles for training. Results on Vaihingen.

more general use-case where one needs to apply the model to a different location.

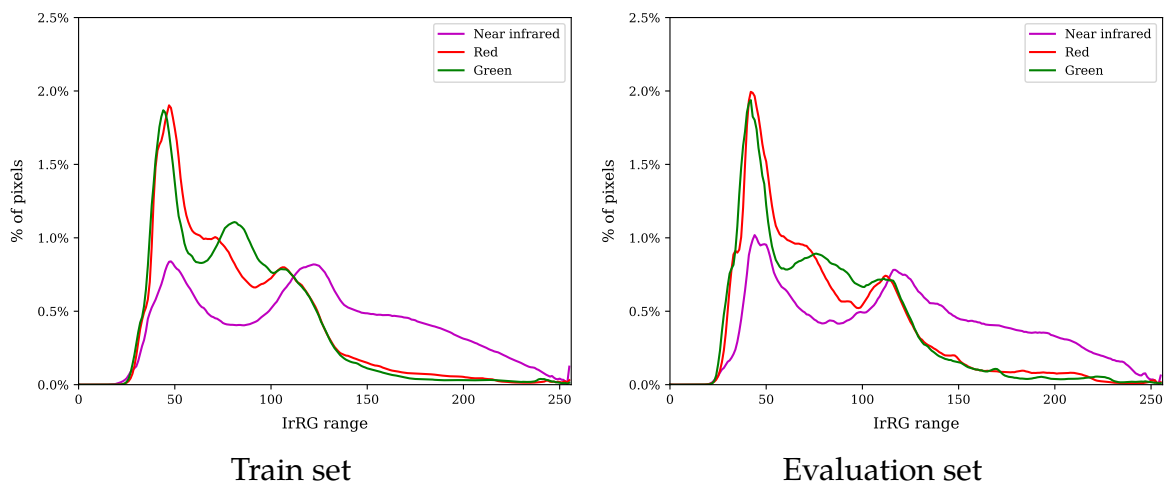


Figure 2.3 – Per channel color histograms over the ISPRS Vaihingen dataset. Comparison between train (left) and evaluation (right) sets.

2.2.2 Supervised learning at large-scale

The previous section stressed out a limitation of standard datasets for semantic mapping in Earth observation. If some already take into account multiple locations, they are devoted to a single class (such as buildings [134, 149]) or to land cover classes [150], but do not offer generic land use classes at a large scale. Consequently, we investigate the generalization capacities of current semantic segmentation methods on a more var-

ied dataset, containing data from different locations and semantically complex land-use classes. To this end, we gather data openly available from different locations –16 cities and their surroundings– in France⁴.

For coherency of comparisons, we use a fixed partition for evaluation, as defined in Table 2.4: 8 cities are used for training and the remaining 8 ones for testing, keeping diversity in terms of architecture and urban design in both subsets. All in all, this new database contains 2121 images, each of them of size $10,000 \times 10,000$ pixels. Therefore, it is 2719 times larger than Vaihingen in terms of surface coverage.

Similarly to Section 2.2.1, we first test the influence of the amount of training data over the classification. However, due to computational times⁵, we conduct more focused experiments. We train with the whole dataset, then only consider 10% of images on the dataset (we make sure to pick 10% of images from each conurbation to conserve the diversity of the dataset), and finally use only one city for training (the seaside town of Caen, which represents a similar amount of data: 12.5% of the entire training set). Test set remains the same.

Table 2.2 – Classification performances with respect to amount of data on large, multi-location dataset.

Train set	OA [%]	mIoU [%]
100 %	52.40	15.79
10 %	50.14	15.25
Caen only ($\sim 12.5\%$)	42.09	10.05

Hence, results are shown in Table 2.2. Performances are not reaching the same level than on Vaihingen, which could be expected since the land use classes are more abstract and difficult than land cover ones and the algorithm is applied to a completely new set of cities (never seen during training). However, considering our current issue, it is worth noting that training with all data or with 10% data leads to similar scores, both in accuracy and IoU. By picking our 10% sample images all over the dataset, we preserved the diversity of the training set and did not degrade the results too much (even if more data is better). On the contrary, training with a single location implies a 10% loss in

4. These data will hereinafter compose what we call *The MiniFrance dataset*. More details about this dataset can be found in Section 2.3

5. Using a Titan X GPU, training over this large dataset takes 40 hours, while testing takes 25 hours

accuracy and 5% less of mIoU. Clearly, in this case, the training set does offer enough variety to encompass all the potential images of the test set.

In a second experiment, we apply the model trained on the whole dataset to each city or conurbation of the test set. Results are shown in Table 2.3. It is interesting to observe that the performance of the network varies significantly between some conurbations, revealing differences between cities and a lack of generalization capacity from the model. Indeed, we observe in Fig. 2.4 that the statistical distribution of the pixel colors differs from train to test. Thus, this large-scale database is a much more diverse dataset than many others, and offers exciting challenges to overcome.

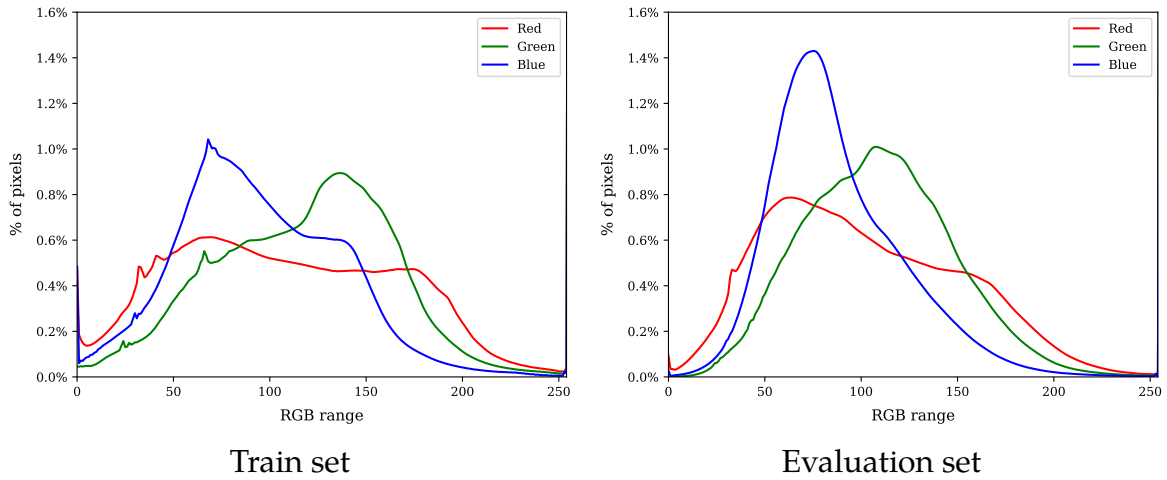




Figure 2.4 – Per channel color histograms over MiniFrance data. Comparison between train (left) and evaluation (right) sets.

Table 2.3 – Performance by conurbation in the multi-location dataset, training over entire train set.

Score	Marseille	Rennes	Angers	Quimper	Vannes	Clermont	Lille	Cherbourg
OA [%]	46.13	51.56	44.85	50.82	49.51	46.51	61.35	67.54
mIoU [%]	12.77	15.05	13.15	13.93	12.66	11.40	16.93	15.82

To better understand these numbers, Fig. 2.5 presents semantic maps obtained during testing. These examples show that the large-scale model performs globally well, yielding reasonable prediction maps. First two rows show quite accurate predictions,

both in a urban scene (first row) and a countryside area (second row). Indeed, the model correctly identifies most classes present on the images. However, the predicted map appears more fragmented than the ground-truth, which shows that the network is sensitive to color variations of the image, and sometimes misses some abstract semantic classes. Third row presents an example where ground-truth is missing for part of the image and the model is still able to correctly classify it as water. Finally, last row shows an example where the network endures some difficulties to distinguish between the *herbaceous vegetation associations*  and *forests* , which demonstrated the challenges presented by these data.

From the experiments performed in Sections 2.2.1 and 2.2.2 we can formulate two main observations:

- First, existing small-scale, mono-site Earth observation datasets, such as the classic ISPRS Vaihingen, are not adapted to measure the generalization capacities of our models to new locations. They are not representative of the life-like scenario where one has annotated data from a certain geographic location, but wants to extend a model over another geographic region. Hence, there is a need for large-scale, multi-location datasets, where train and evaluation sets come from different locations, and where classes have high appearance variability.
- Second, as the experiments on the multi-location, large-scale setting have shown (Section 2.2.2), current supervised semantic segmentation approaches are not able to generalize correctly –nor homogeneously– to unseen locations. How can we evolve to more robust and generic models?

In real-life Earth observation applications, usually one has access to a small portion of annotated data (typically coming from one geographic location), while there is a plethora of non annotated data at disposal. We believe that these unlabeled data are essential to fill the gap of generalization. Therefore, we strongly believe that semi-supervised models –which leverage unlabeled data to help on the learning process– should be studied in more detail.

In consideration of all of the above, we present in the following section the MiniFrance suite, the first dataset especially designed to evaluate and compare semi-supervised semantic segmentation methods in Earth observation. Its composition is inspired from the previous experiments. On the one hand, MiniFrance includes labeled and unlabeled data for training, as in real-life applications. On the other hand, labeled data come from only a few cities, which represents the more challenging setting for learning and gen-

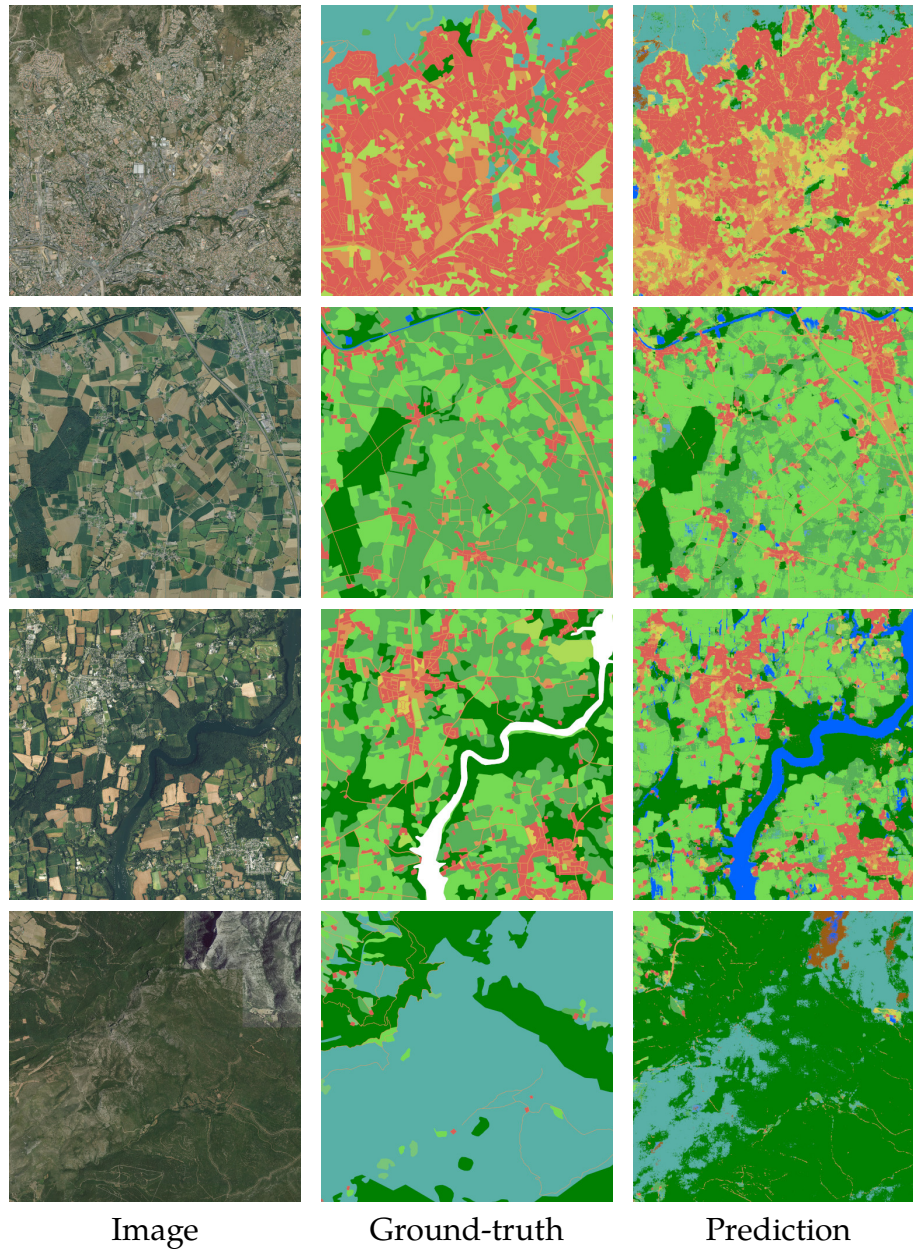


Figure 2.5 – Semantic segmentation results on multi-location data. Legend of main classes: Urban fabric ■; Industrial, commercial, public, military, private and transport units ■; Arable land ■; Pastures ■; Herbaceous vegetation associations ■; Forests ■; Water ■.

eralize (see Table 2.2, Caen only results) and unlabeled data come from several cities to add variety to the training set (Table 2.2, 10% of training data shows that variety is key to seize the important features of images). Moreover, the test data are chosen to be geographically independent from training data, simulating a real application scenario.

2.3 The MiniFrance suite

Considering the limitations of current Earth Observation (EO) datasets emphasized and evinced in Sections 2.1 and 2.2, we propose a new large-scale benchmark suite for semi-supervised semantic segmentation: MiniFrance. As in real life EO applications, it comprises both labeled and unlabeled imagery for developing and training algorithms. To our knowledge, this is the first dataset designed for benchmarking semi-supervised learning in the field. Moreover, it consists of a variety of classes on several locations with different appearances: this opens the opportunity to push further the generalization capacities of the models.

2.3.1 MiniFrance

It consists of data corresponding to 16 conurbations and their surroundings from different regions in France (see Figure 2.6 and Table 2.4). It includes urban and countryside scenes: residential areas, industrial and commercial zones but also fields, forests, sea-shore or low mountains.

MiniFrance gathers data from two sources:

- Open data VHR aerial images from the French National Institute of Geographical and Forest Information (IGN) BD ORTHO database⁶. They are provided as RGB tiles of size 10,000 px × 10,000 px at a resolution of 50 cm/px, namely 25 km² per tile. Images included in this dataset were acquired between 2012 and 2014.
- Labeled class-reference from the UrbanAtlas 2012 database. Original data are openly available as vector images (i.e. containing polygon annotations) at the European Copernicus program website⁷. Using the georeferenced data available in the BD ORTHO, we have made rasters of these images that geographically match the VHR

6. <https://geoservices.ign.fr/documentation/diffusion/index.html>

7. <https://land.copernicus.eu/local/urban-atlas/urban-atlas-2012/view>

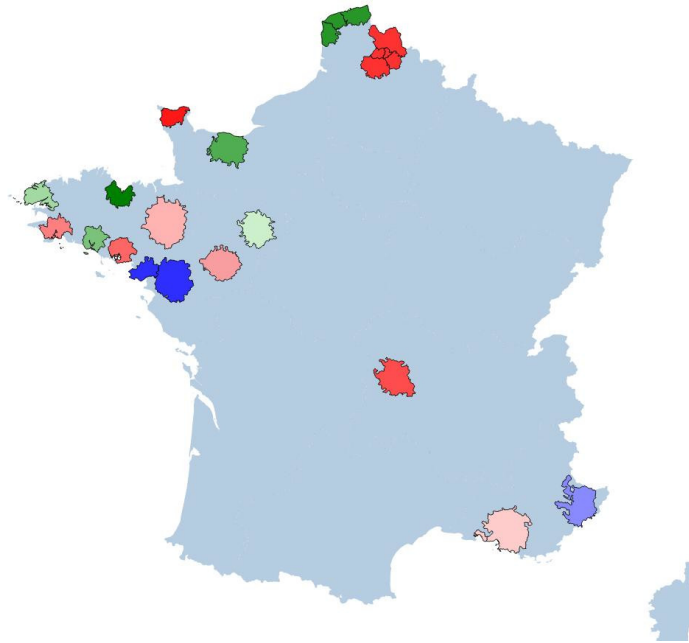


















Figure 2.6 – MiniFrance dataset overview. Areas colored according to Table 2.4.

Table 2.4 – List of cities in MiniFrance and split details.

		<i>Conurbation</i>	<i>Tiles</i>	<i>% pixels</i>	<i>Color</i>
<i>Training</i>	<i>Labeled</i>	Nice	170	8.01 %	
		Nantes, Saint-Nazaire	226	10.65 %	
	<i>Unlabeled</i>	Le Mans	107	5.04 %	
		Brest	88	4.14 %	
		Lorient	68	3.20 %	
		Caen	126	5.94 %	
		Dunkerque, Calais, Boulogne-sur-Mer	150	7.07 %	
		Saint-Brieuc	71	3.34 %	
	<i>Test</i>	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, Hénins		275	12.96 %		
Cherbourg		57	2.68 %		

tiles from the BD ORTHO. We consider 14 land-use classes (see Table 2.5), corresponding to the second level of the semantic hierarchy defined by UrbanAtlas [151]. For this reason, some of them might not be present in the regions considered for MiniFrance and they are colored in gray in Table 2.5.

Collecting data from different sources brings some burden that must be considered. Land use maps from UrbanAtlas are obtained through a semi-automatic process and thus they are not 100% accurate [152], besides polygon annotations might not match 50 cm/px resolution images precisely. Moreover, additional errors might come from the fact that image and ground-truth may not correspond to the same year. Nonetheless, MiniFrance has several peculiar, unprecedented properties that we detail now.

Large-scale. MiniFrance is a very large-scale dataset. It contains a total of 2,121 aerial images of size 10,000px × 10,000px at 50cm/px resolution. In terms of ground coverage, with 53,000 km² it is 12 times larger than DeepGlobe and larger than xBD, among the datasets of similar resolution.

Rich and varied. MiniFrance includes aerial images of 16 conurbations and their surroundings from different regions with various climates and landscapes (Mediterranean, oceanic and mountainous) in France. Introducing various locations leads to various appearances for the same class (buildings look different, vegetation is not the same and so on). Moreover, it combines urban centers, rural areas and large forest scenes. With respect to remote sensing datasets like ISPRS Vaihingen and Potsdam, it offers much more variety, as already observed in Section 2.2.

High semantic level of classes. MiniFrance considers 14 land-use classes, which is more than most of the datasets exposed in Section 2.1. However, these classes have higher semantics: to identify an “urban area”, an algorithm must be able to find several houses or buildings together, same to classify a forest. It is much easier to only consider classes at an object level (cars, buildings, trees, etc). Moreover, land-use classes are hard to learn, even for humans: how to distinguish *pastures* ■ from *artificial non-agricultural vegetated areas* ■ in Figure 2.7?

Underlying domain adaptation problem. Since train and test sets were split by city – instead of excluding random tiles from all the zones– algorithms developed on MiniFrance must address the underlying problem of domain adaptation. The appearance of classes might vary considerably from one city to another. Architecture is not the same, agriculture may change, etc. In Figure 2.7 we observe that *urban fabric* ■ does not look alike between the three exposed images.

Table 2.5 – Land use classes available in MiniFrance.
















<i>Class</i>	<i>% pixels</i>	<i>Color</i>
Urban fabric	9.6 %	
Industrial, commercial, public, military, private and transport units	6.4 %	
Mine, dump and construction sites	0.7 %	
Artificial non-agricultural vegetated areas	1.1 %	
Arable land (annual crops)	29.5 %	
Permanent crops	1.0 %	
Pastures	29.0 %	
Complex and mixed cultivation patterns	0.0 %	
Orchards at the fringe of urban classes	0.0 %	
Forests	15.9 %	
Herbaceous vegetation associations	4.6 %	
Open spaces with little or no vegetation	0.4 %	
Wetlands	0.7 %	
Water	1.0 %	
Clouds, shadows or no data	0.1 %	



Figure 2.7 – Some samples of MiniFrance dataset on different localizations. Images (up) and their associated ground-truth (down). From left to right: Nice, Rennes and Vannes.

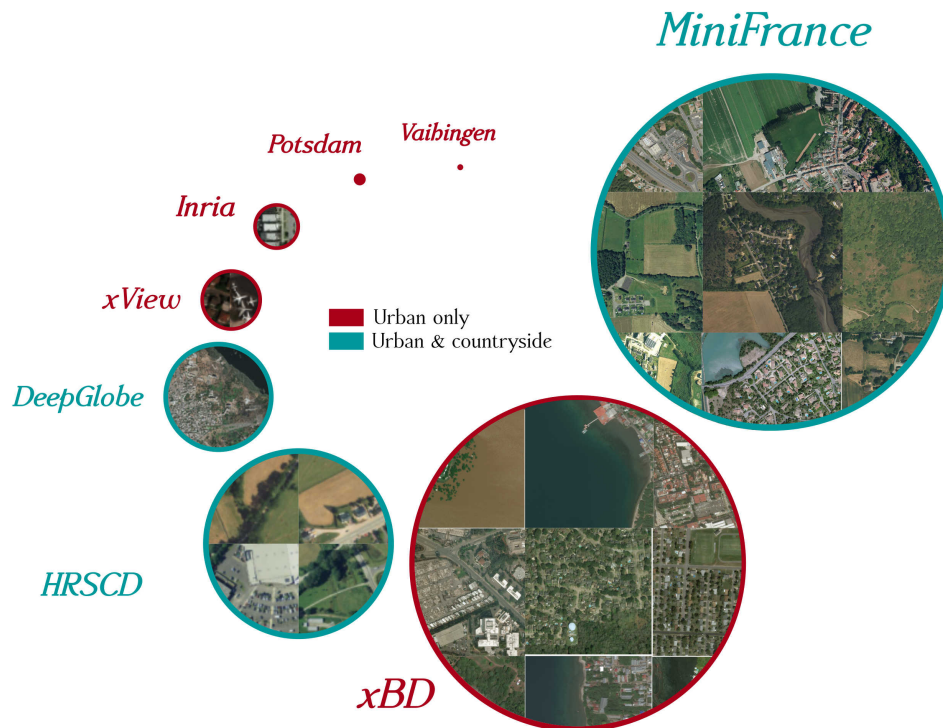


Figure 2.8 – Representation of public EO datasets. Circle surface is proportional to real surface coverage. MiniFrance covers an important surface, contains images from urban and countryside scenes, and is the only dataset designed for benchmarking semi-supervised methods.

Designed for semi-supervised semantic segmentation. To our knowledge, this is the first dataset specifically designed for semi-supervised learning strategies. Indeed, our training split includes labeled (two cities) and unlabeled images (six ones) while algorithms can be tested on the eight remaining cities. Such a proportion of unlabeled examples fosters the development of new methods to leverage them. Moreover, these methods are likely to be easily transferred to lifelike scenarios and to have better generalization properties by design. Table 2.4 presents our training –labeled and unlabeled images– and testing splits.

Fig. 2.8 shows a graphical comparison of MiniFrance with other well-known Earth observation benchmarks at similar resolution.

Finally, the MiniFrance dataset is publicly available and can be downloaded from [IEEE Dataport](https://iee-dataport.org/open-access/minifrance)⁸.

8. <https://iee-dataport.org/open-access/minifrance>

2.3.2 TinyMiniFrance

With the purpose of prototyping new algorithms with fast processing and validation times, we also introduce *tinyMiniFrance* (tMF), a small, computationally tractable version of the MiniFrance dataset.

Our *tinyMiniFrance* consists in a subsample of the original data: it contains 3,500 images of size $1,000\text{px} \times 1,000\text{px}$. Containing around 1.7% of the original data, it preserves the variety and richness of MiniFrance.

Sampling is uniform over each region. To preserve the same balance between classes, it is performed by randomly selecting sub-tiles from original tiles in the dataset and verifying that there is at least one sub-tile from each tile in MiniFrance. Figure 2.9 illustrates the result of sampling over the region of Cherbourg. Moreover, we keep the original proportion of images per region on the dataset (e.g. the region of Nice contains more data than Brest, as in Table 2.4). Training –labeled and unlabeled– and testing splits remain unchanged with respect to the original dataset.

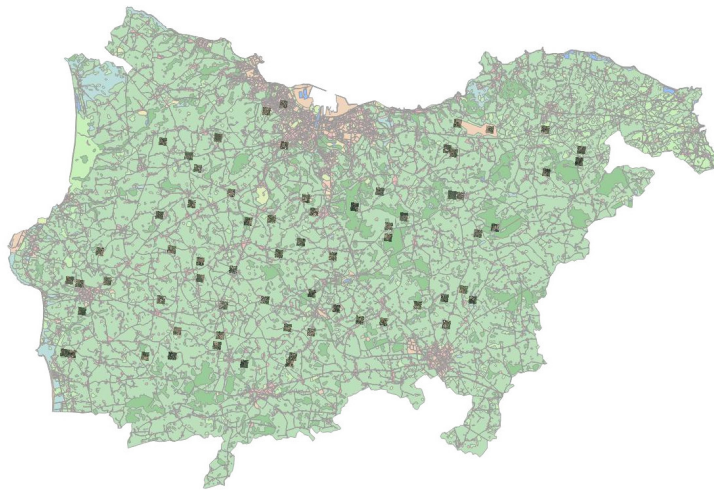


Figure 2.9 – Subsample for *tinyMiniFrance* over Cherbourg region.

Table 2.6 shows the classes distribution over *tinyMiniFrance*. When compared with Table 2.5, the original proportions of classes of MiniFrance are well preserved. Thus, we can expect that algorithms developed on *tinyMiniFrance* will scale up similarly to MiniFrance. For this reason and for computing capacities, all the following analysis will be performed over *tinyMiniFrance* (Section 2.4). For the sake of simplicity, we will mostly employ the term MiniFrance.

Table 2.6 – Classes distribution on tinyMiniFrance.

<i>Class</i>	<i>% px</i>	<i>Class</i>	<i>% px</i>	<i>Class</i>	<i>% px</i>
Urban	9.9 %	Permanent	1.3 %	Herbaceous	4.5 %
Industrial	6.5 %	Pastures	27.3 %	Open	0.1 %
Mine	0.7 %	Complex	0.0 %	Wetlands	0.7 %
Artificial	1.2 %	Orchards	0.0 %	Water	1.0 %
Arable	30.7 %	Forest	16.0 %	Clouds	0.1 %

2.4 Statistical analysis of the representativeness of training and test datasets

This section introduces two concepts that are required to have adequate learning conditions to achieve satisfying results and that explain our choice for labeled training data, unlabeled training data and test data for MiniFrance: class representativeness and appearance.

On the one hand, class representativeness refers to the fact that to properly learn a certain class, any learning algorithm needs to see at least some examples of this class during training. Otherwise, it will not be able to identify it successfully at inference time. Hence, the labeled training split should contain examples of all possible classes in the dataset.

On the other hand, in a standard supervised setting, appearance features in the training set should have the same distribution as those on the test set to achieve good inference results. However, in a semi-supervised learning setting, unlabeled training data relax such a strong constraint. Indeed, by providing more information on the possible visual features, they help learning a wider appearance of each class. This is appealing since it favors generalization, but also brings more robustness against distribution shift (i.e. it is more unlikely that the test set contains very new appearances w.r.t. the test set).

According to this, we consider that a good training split should satisfy two conditions:

- (i) Labeled training data must contain a good representation of all classes in the dataset, ideally with the same distribution than the testing data.
- (ii) Training data (labeled and unlabeled) must cover all the range of appearances of different visual features in the dataset.

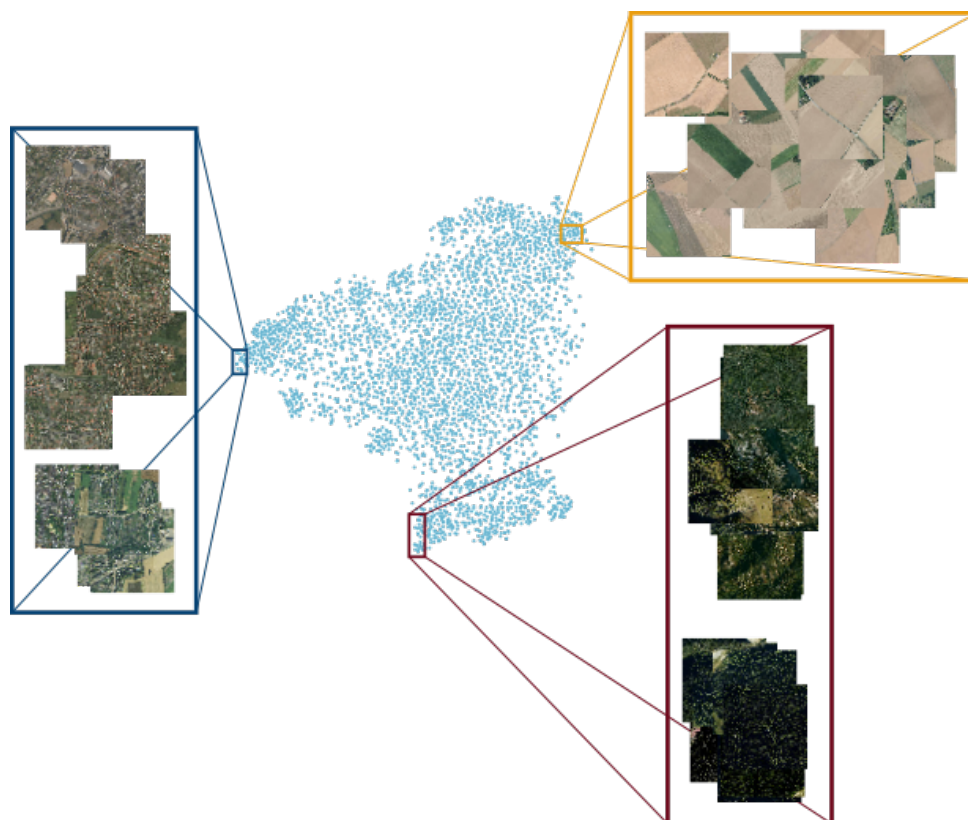


Figure 2.10 – 2D representation of images by t-SNE after ResNet34 encoding. Similar projections are close, while different visual features are separated. In ■, mostly urban scenes; in ■ fields images and in ■ mostly forest scenes.

In what follows we present a statistical analysis of the MiniFrance dataset to show that our chosen split (in Table 2.4) satisfies these two requirements.

2.4.1 Appearance analysis

To study the appearance similarity between the training split and testing split of multi-locations datasets –such as MiniFrance–, we rely mainly on three tools.

First, we use pre-trained Convolutional Neural Networks (CNNs) as image feature extractors. Indeed, thanks to their shared-weight architecture and translation invariance, CNNs are reliable encoding tools for images. Furthermore models pretrained on ImageNet –a very large database for visual recognition– have seen a wide variety of representations that allow them to output a vector encoding the image’s appearance.

Second, we make use of the t-SNE [153] algorithm to reduce the dimension of high-

dimensional feature vectors and visualize them in a 2D space. t-SNE is a non-linear dimensionality reduction technique that allows visualization of high-dimensional data. In brief, the algorithm starts by converting the Euclidean distances between high dimensional objects into conditional probabilities that represent similarities. Then, it defines a Student t-distribution with one degree of freedom over the low-dimensional points. Finally, it minimizes the Kullback-Leibler divergence between the high and low-dimensional distributions with respect to the locations of the low-dimensional points. At the end, if two high-dimensional objects are similar, then their representations at the low-dimensional t-SNE visualization are close and vice-versa.

Third, we rely on the one-class SVM algorithm [154] to estimate the support of the data distributions. In a nutshell, this algorithm uses a support vector machine to separate all the data points from the origin (in a feature space), by maximizing the distance from this hyperplane to the origin. As a result, one obtains a binary function that captures regions in the input space where the probability density of the data lives.

Thus, our algorithm for appearance coverage assessment between datasets is summarized as follows:

Step 1. For each image in the dataset, we obtain an encoded feature vector through a CNN (in particular, we use a VGG16 [155] and a ResNet34 [58]).

Step 2. Then, we apply a t-SNE to this set of high-dimensional feature vectors to obtain a 2D representation of the dataset images which preserves the original similarity of visual features.

Step 3. Each point in the 2D space can be traced back to the original tile and so to the city it comes from. Then, we use a one-class SVM [154] to estimate the distribution of the city images in the 2D space.

Step 4. Finally, we evaluate the appearance similarity and coverage between cities using two metrics:

(i) We use the intersection over union score (IoU, the standard metric for object detection) between the surfaces defined by the distributions, or appearance maps, to assess appearance similarity. Let S_1 and S_2 be two sets, the IoU score between them is defined as $IoU(S_1, S_2) = \frac{|S_1 \cap S_2|}{|S_1 \cup S_2|}$. In our context, higher IoU scores relate to resemblance between the appearance maps of cities.

(ii) We also introduce the Intersection over Test area score (IoT). Let S_1 and S_2 be two sets, the IOT score between them is defined as $IoT(S_1, S_2) = \frac{|S_1 \cap S_2|}{|S_2|}$. This score

measures the area covered by the intersection of the two surfaces normalized over the second area, which is the objective. We compute IoT considering $S_1 \in T$ and $S_2 \in E$, where T and E are the set of training cities and the set of testing cities, respectively. Thereby IoT measures how well the objective appearance map is covered by appearances of the training data.

Fig. 2.11 presents a visualization that summarizes the algorithm for appearance similarity assessment.

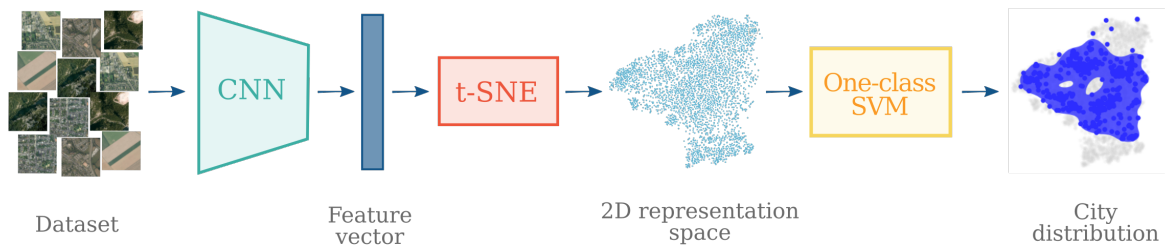


Figure 2.11 – Generation of appearance maps for multi-location image datasets. First, we encode images in a dataset as high-dimensional feature vectors by the means of pre-trained CNNs. Then, the t-SNE algorithm is applied to reduce the dimension of the feature vectors to a 2D-space. Given the assumption that CNNs encode for image appearance, look-alike images should be close in the 2D representation space, while images with different visual features should be apart. Finally, one-class SVM is applied over data points coming from the same location to estimate the data distribution of each specific location.

Furthermore, we apply this method to the MiniFrance data. The results presented in this section used a ResNet34 encoding for MiniFrance images, even though VGG16 encoding yields similar outcomes (step 1). Fig. 2.10 shows the mapping resulting after application of the t-SNE algorithm (step 2), it validates that similar images are close while different appearances are put apart. Results of step 3 are shown in Fig. 2.12, that shows the appearance maps obtained for each city in the dataset. Finally, Fig. 2.13 shows IoU and IoT scores as two heatmaps between cities in the training set and the ones in the test set.

Results are consistent with reality, to name a few examples: Nice exhibits low similarity scores with all cities, except Marseille, because those are the only cities from Provence, on the Mediterranean coast. Quimper has its higher IoU score with Brest,

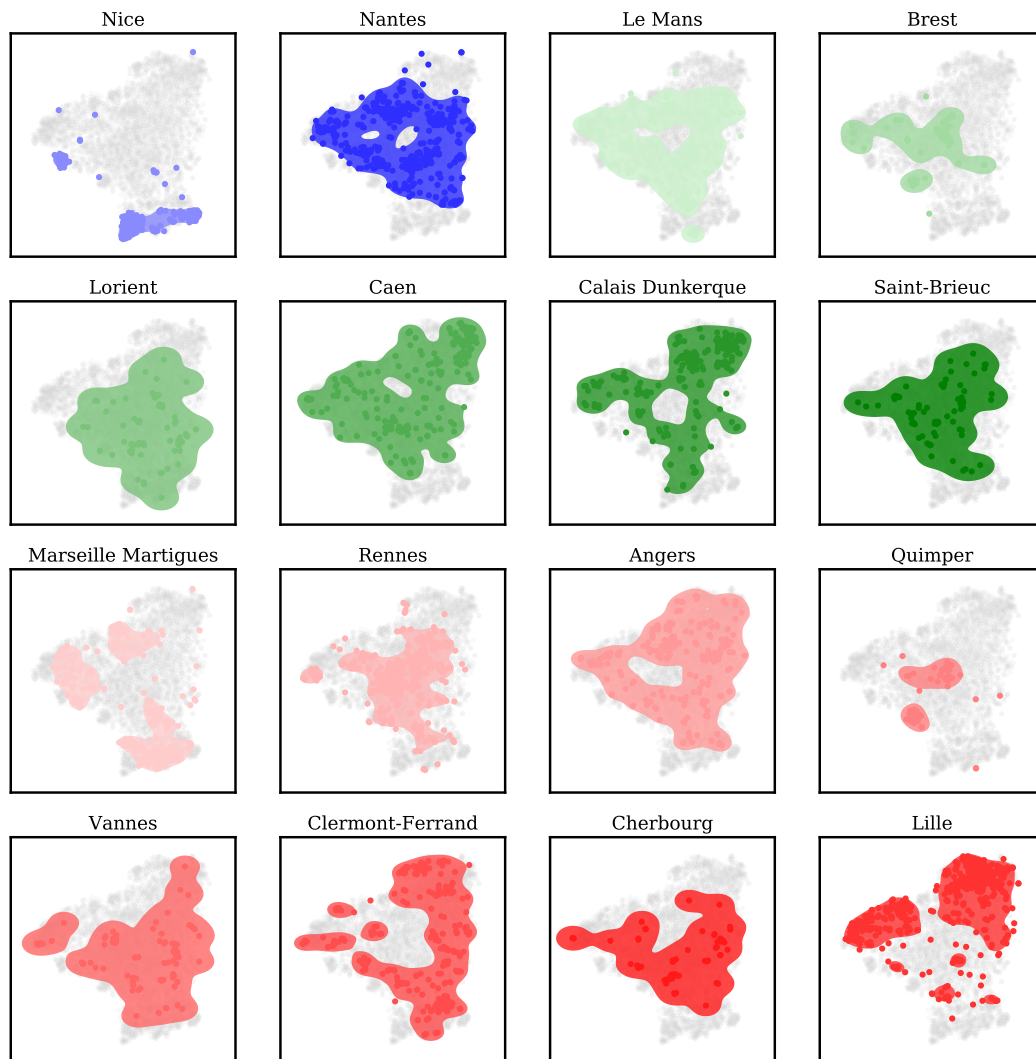


Figure 2.12 – Distributions of cities in the 2D appearance space.

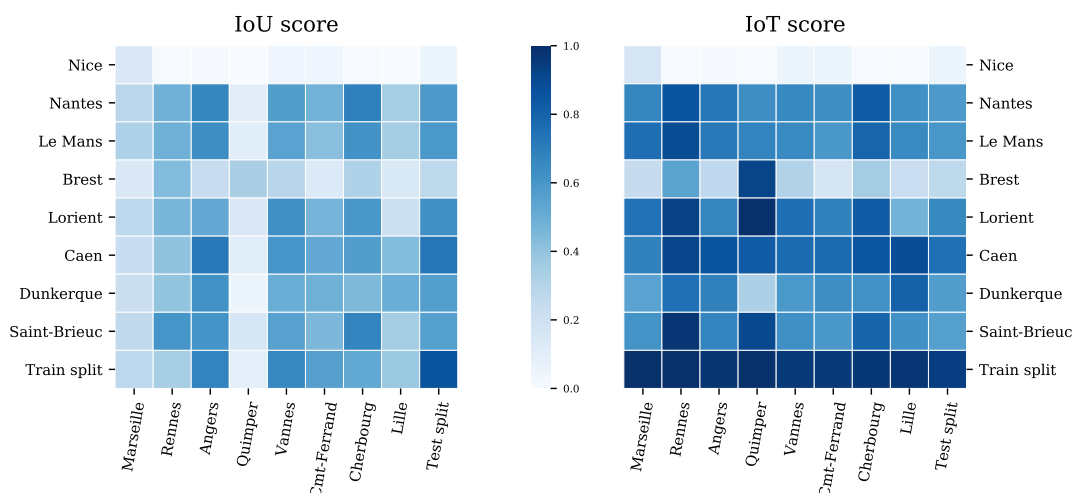


Figure 2.13 – IoU and IoT (Intersection over Test) scores between the 2D distributions of cities in the training split and the testing split, represented as heatmaps. Last column represents the scores between a training city and the union of surfaces of the testing split. Similarly, last row corresponds to the scores between the union of surfaces in the training split and every city in the test. The dark last row of the IoT score indicates that the train split covers well every city in the test partition.

which is coherent because of their geographic proximity; in terms of IoT Quimper is well covered by Lorient and Saint-Brieuc, which are also geographically close (all these cities are located in Brittany). High IoU score between Angers and Caen is justified by the fact that both are agricultural localities, with similar landscapes.

To summarize, this section proposes a method to assess representativeness in terms of appearance similarity between sites on multi-location datasets. In particular, we apply this tool to the MiniFrance data, which allows us to perform a comparison between cities in the training split and the ones in the testing split. IoU scores show that, even if there are similarities between cities, no locality in the training set is identical to another one in the test set. However, IoT proves that testing cities are well covered by the ensemble of training cities, which is confirmed by the last dark row of this score in Figure 2.13 (right).

2.4.2 Class representativeness analysis.

In this section, we present two tools to assess class representativeness: class distribution histograms and class spatial distribution maps.

An underlying assumption of ML is data distribution stationarity between learning and inference time, that is a class cannot be learnt if no example of it has been seen at training time. In other words, the labeled training partition has to contain all the existing classes on the dataset. If possible, the distribution of the classes during training should be similar to the one of test data.

To fulfill this condition, we study the classes distribution on a multi-location dataset by the means of two tools:

Tool 1. Class histograms for each location on the dataset.

Tool 2. The 2D-representation space obtained from the appearance assessment algorithm (see Section 2.4.1) allows us to perform an analysis of the distribution of classes over the images in terms of appearance.

In particular, we apply these tools to the MiniFrance data.

We compute class histograms of each geographic area (tool 1) and present them in Fig. 2.14. We observe that they vary significantly from one city to another. Besides, among the 12 classes that we consider in this analysis –we do not consider *complex and mixed cultivation patterns, orchards at the fringe of urban classes* nor *clouds and shadows*⁹, see Table 2.5–, no city contains all of them. The best coverage of classes is given by the *Nantes, Saint-Nazaire* or *Marseille, Martigues* conurbations that exhibit 10 of the classes. However, most of the regions contain only 7 or 8 categories in total.

Another problem is the heterogeneous proportions of classes in each region. The most striking example is *Cherbourg* where 6 classes are represented and one of them –*pastures*– covers 70% of the total pixels, while the other categories count for less than 10% each.

Therefore, defining a labeled training split that represents all the classes in a good proportion is not straightforward.

9. *clouds and shadows* is not a land use class and thus it is not interesting in our case.

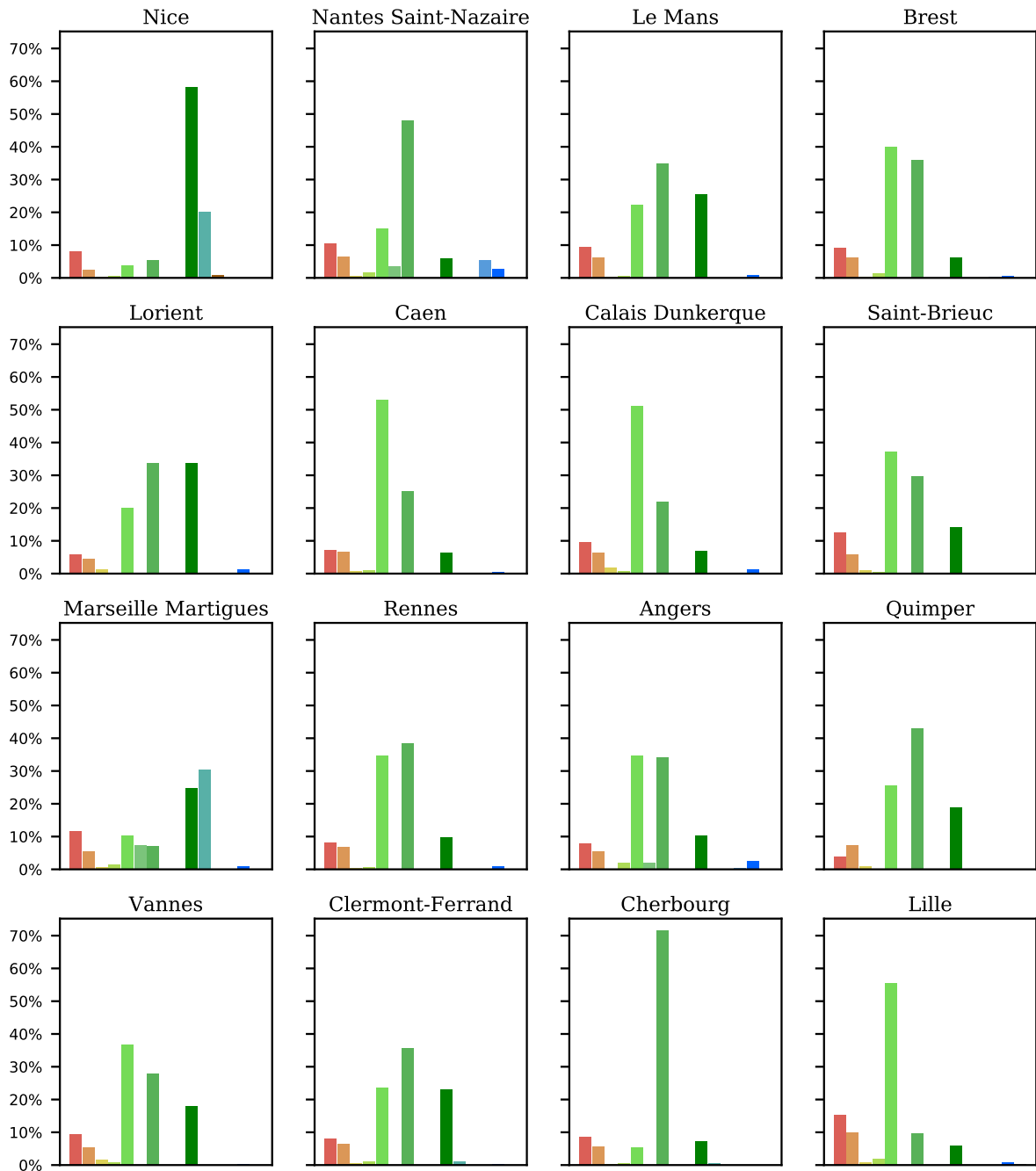


Figure 2.14 – Histograms of class distributions by city. x axis represents the classes with colors as in Table 2.5. y axis presents the percentage of each class by city.

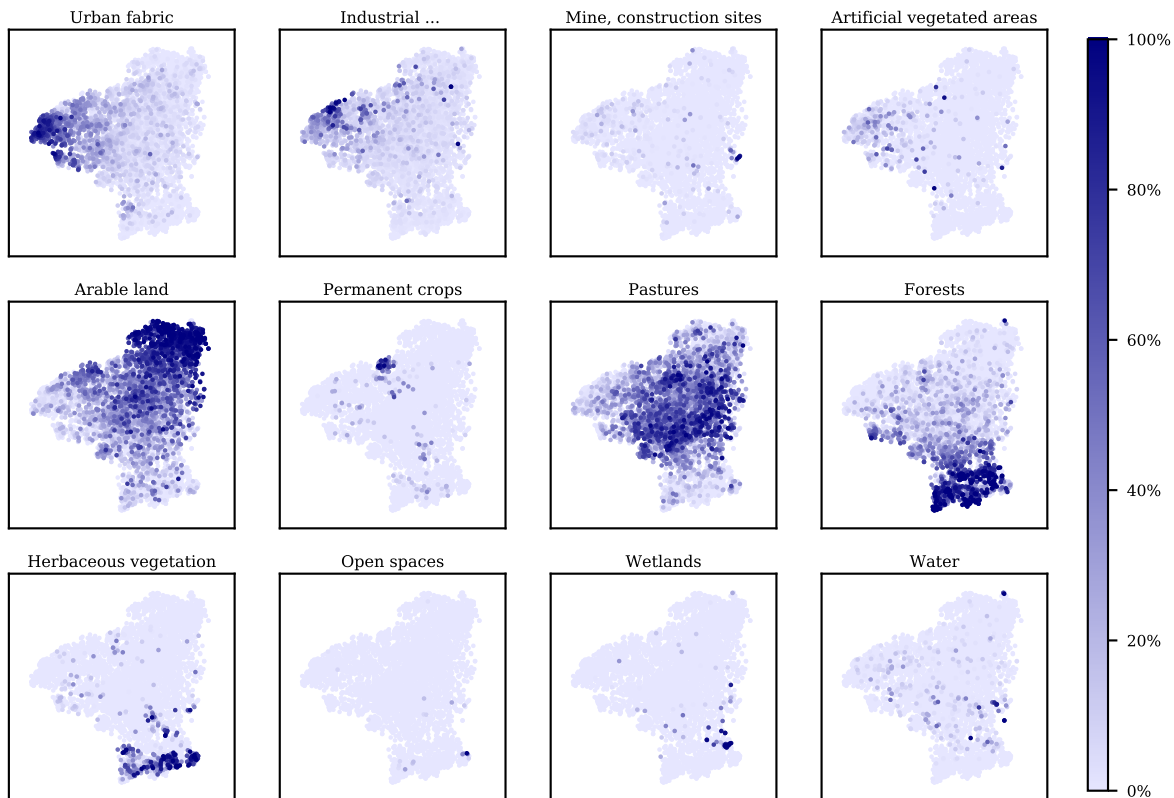


Figure 2.15 – Class distributions in the 2D appearance space. One subplot represents one class. Each point is colored as the proportion occupied by a given class over the corresponding image.

Along with the histograms, we make use of our precedent analysis to understand the distribution of the classes in terms of image appearance (tool 2). Each subplot in Figure 2.15 presents a class in the dataset and contains all the images in the 2D appearance representation space. Each point is colored according to the proportion occupied by the class over the image. That is, the darker the point in the figure ■, the more pixels corresponding to the class are in the image. On the contrary, a light point ■ indicates that there are very few pixels representing the class. We observe that some classes (such as *pastures* or *arable land*) are well-spread over the whole appearance space, with high proportions in many tiles. This means that they are represented by diverse images and that they are likely to have a lot of examples (as confirmed by the histograms of Fig. 2.14). These classes should be easier to learn. Others –like *urban fabric* or *industrial, commercial, public, military, private and transport units*– are widespread, but do not reach majority in most of the images in which they are present. This means that these classes have a large

variance in their appearance but not so many examples per appearance mode, which could make them more difficult to learn. Moreover, other categories (like *artificial non-agricultural vegetated areas* or *herbaceous vegetation associations*) are mostly concentrated over one zone –that could correspond to only one geographic region–, that is, they are present in images of specific appearances, which makes them even harder to learn. Finally, we see classes that are extremely rare (e.g. *wetlands* and *open spaces with little or no vegetation*), they are present in a few images only, and thus they should be the more difficult to learn.

Sections 2.4.1 and 2.4.2 have shown that we can combine class distribution and visual appearance mapping to get further insight on the data. These tools help us to define a suitable partition of the MiniFrance dataset –labeled, unlabeled and test data– that satisfies the class distribution and appearance conditions as we will show in Section 2.5.

2.5 Defining the labeled, unlabeled and test splits for MiniFrance

Using all the tools and information presented in Section 2.4, MiniFrance has been carefully designed to satisfy the conditions of appearance and class representativeness, that make it appropriate for semi-supervised learning. Indeed, the split proposed in Table 2.4 allows us to represent all the classes with a proper distribution, as shown in the histograms of Figure 2.16. Hence, all classes present in the test set have training examples in the labeled split.

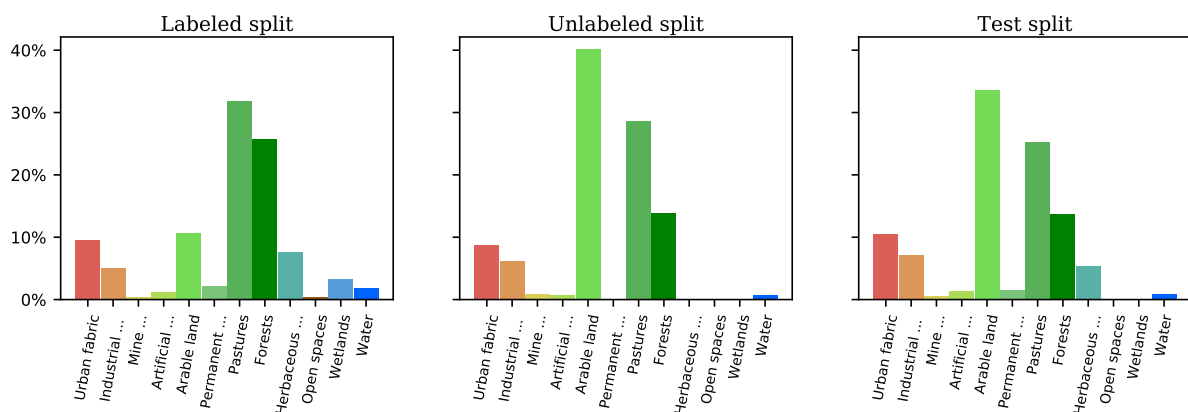


Figure 2.16 – Class distributions aggregated by split as defined in Table 2.4.

On the appearance side as shown in Figure 2.17, even if labeled cities do not cover the whole appearance space of test images, the union of labeled and unlabeled does. This should ensure that all appearances are seen in a semi-supervised setup. Moreover, in terms of IoU scores of appearance shown in Figure 2.13, the labeled split comprises one region with a high score (Nantes) and one with a low score (Nice) which should help to learn different appearances of classes. In addition, in the unlabeled split most of the cities have a high score with respect to the test set, so they should help to extract the implicit information from images.

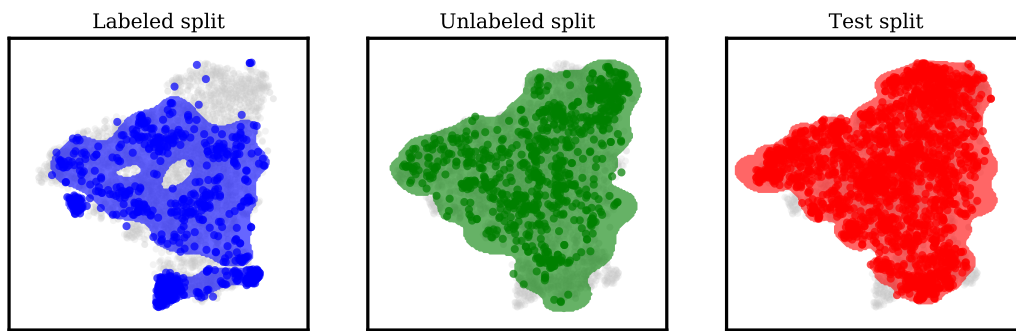


Figure 2.17 – Appearance representation aggregated by split as defined in Table 2.4.

Table 2.7 – IoU and IoT scores between training data –labeled and unlabeled– and test data. Scores are presented in numerical form as well as color code for comparison with Figure 2.13.

$S_1 - S_2$	$IoU(S_1, S_2)$	$IoT(S_1, S_2)$
Labeled - Test	0.63 ■	0.64 ■
Unlabeled - Test	0.87 ■	0.93 ■

Table 2.7 presents the IoU and IoT scores between the surfaces in Figure 2.17 and confirms the information above. Thus, even if the labeled training split contains all classes of the test split, 64% of IoT means it is far from covering all the possible appearances. However, with 93% of IoT score with the test area, the unlabeled training split offers wider information about the visual features present in the MiniFrance dataset that should be exploited to achieve good quality classification and generalization.

In brief, MiniFrance is a very challenging dataset for semantic segmentation that

promotes new solutions in a semi-supervised manner as some appearances can only be extracted from the unlabeled data. However, train and test adequacy was carefully controlled to avoid domain shift and such disentangle semi-supervised learning from domain adaptation and transfer learning.

2.6 Comparing MiniFrance to classic datasets

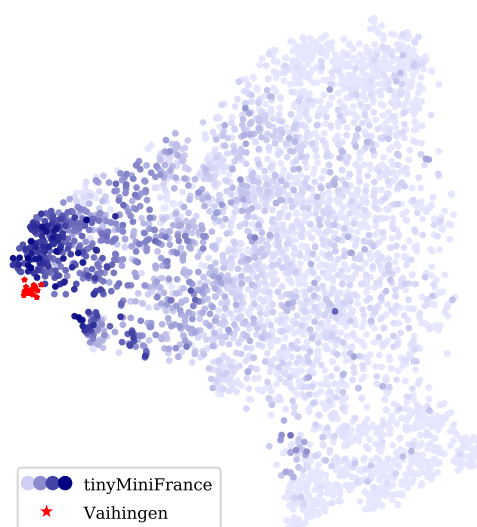


Figure 2.18 – 2D representation of images by t-SNE, applied to tinyMiniFrance and Vaihingen together, after ResNet34 encoding. Points from tinyMiniFrance are colored according to the proportion occupied by the class *urban fabric*.

Finally, in order to validate the fact that MiniFrance is more challenging in terms of appearance and variability than more classic datasets, we apply our tool for appearance coverage assessment previously presented (Section 2.4.1) to the union of both datasets, tinyMiniFrance and Vaihingen. To get a fair comparison, images from the Vaihingen dataset were downsampled to the tinyMiniFrance resolution (from 9 cm/px to 50 cm/px) before being encoded by the CNN.

Due to the stochastic nature of the t-SNE algorithm, it is important to note that subsequent runs can lead to different embeddings. However since tinyMiniFrance is much larger than the 16 Vaihingen tiles, the projection is not noticeably perturbed up to rotation and reflection. We chose the embedding which resulted in the same visualization

as Section 2.4.

Results are shown in Figure 2.18. Red stars (\star) represent Vaihingen tiles, while shading blue circles ($\bullet \cdots \bullet$) are tinyMiniFrance tiles, colored according to the proportion occupied by *urban fabric* (as in Figure 2.15, darker points contain a higher proportion of urban pixels). We consider specifically the *urban fabric* class since it is the most related to the Vaihingen urban dataset.

The previous visualization is insightful. On the one hand, we realize how small the Vaihingen dataset is compared to tinyMiniFrance (and even more to the entire MiniFrance), in terms of number of available tiles. On the other hand, the t-SNE algorithm places Vaihingen as a very small cluster next to the urban scenes of tinyMiniFrance, which means that:

- (i) Vaihingen is slightly different from tinyMiniFrance (maybe due to the IRRG encoding vs. RGB);
- (ii) At the same time, Vaihingen remains visually close to the urban images from tinyMiniFrance (confirming our choice to consider here the *urban fabric* class); and
- (iii) The wide surface covered by tinyMiniFrance on the 2D appearance projection space w.r.t. Vaihingen shows that our dataset presents a much larger variety of appearances in terms of urban scenes; furthermore, these urban scenes form only a small part of the appearance space, thus proving the very wide diversity of tinyMiniFrance, and to a larger extent of MiniFrance.

2.7 Data fusion contest 2022: MF-DFC22

As we previously mentioned, every year since 2006 the IEEE Geoscience and Remote Sensing Society (GRSS) has organized the Data Fusion Contest, aiming to promote theoretical advances and best practices in image analysis and data fusion for remote sensing applications.

Co-organized by the Image and Data Fusion Technical Committee (IADF TC) of the IEEE GRSS, ONERA, Université Bretagne Sud and ESA ϕ -lab, **the 2022 IEEE GRSS Data Fusion Contest (DFC22) is about semi-supervised learning**. It aims to foster research in automatic land cover classification from only partially annotated training data, by leveraging large amounts of unlabeled data.

To this end, the DFC22 is based on MiniFrance (Section 2.3). Indeed, the MiniFrance-DFC22 (MF-DFC22) dataset extends and modifies the MiniFrance dataset for training semi-supervised semantic segmentation models for land use/land cover mapping. The multimodal MF-DFC22 contains aerial images, elevation model, and land use/land cover maps corresponding to 19 conurbations and their surroundings from different regions in France, gathering data from three sources:

- Open data VHR aerial images from the French National Institute of Geographical and Forest Information (IGN) BD ORTHO database. They are provided as 8-bit RGB tiles of size $\sim 2,000\text{px} \times \sim 2,000\text{px}$ at a resolution of 50cm/px, namely 1 km² per tile. Images included in this dataset were acquired between 2012 and 2014.
- Open data Digital Elevation Model (DEM) tiles from the IGN RGE ALTI database. DEM data give a representation of the bare ground (bare earth) topographic surface of the Earth. They are provided as 32-bit float rasters of size $\sim 1,000\text{px} \times \sim 1,000\text{px}$ at a spatial resolution of 100cm/px, i.e. also 1 km² per tile. The altitude is given in meters, with sub-metric precision in most locations. This database is regularly updated so images included in the dataset were acquired between 2019 and 2020.
- Labeled class-reference from the UrbanAtlas 2012 database. 14 land-use classes are considered, corresponding to the second level of the semantic hierarchy defined by UrbanAtlas. Original data are openly available as vector images at the European Copernicus program website and were used to create raster maps that geographically match the VHR tiles from BD ORTHO. They are provided as integer rasters with index labels (0 to 15 –8 and 9 being UrbanAtlas classes which do not appear in the regions considered–) of size $\sim 2,000\text{px} \times \sim 2,000\text{px}$ at a resolution of 50cm/px, namely 1 km² per tile.

Slightly different from the original MiniFrance, the MF-DFC22 dataset is organized as follows:

The training partition (labeled + unlabeled, same areas from MiniFrance, as detailed in Table 2.4), contains a total of 1915 tiles. The largest area corresponds to Nantes/Saint-Nazaire with 433 tiles, while the smallest area is Lorient with only 120 tiles. Data are provided with georeference information.

The validation partition contains eight georeferenced areas corresponding to the testing partition of MiniFrance (Table 2.4), with RGB images and DEM information. This

partition contains 2066 tiles. Largest area is Lille/Arras/Lens/Douai/Henin including 407 tiles, smallest one is Cherbourg with 113 tiles.

The test partition consists of three areas without georeference information and contains RGB images and DEM information only. This partition includes 1035 tiles.

With two possible tracks to participate –track 1 on semi-supervised land cover mapping and track 2 on brave new ideas–, the DFC22 was officially launched on January 3rd, 2022. Winners will be announced on March 25th. The MF-DFC22 data are openly available at [DFC22 IEEE Dataport](https://iee-dataport.org/competitions/data-fusion-contest-2022-dfc2022)¹⁰. Finally, classification results will be submitted to the [Codalab competition site](https://codalab.lisn.upsaclay.fr/competitions/880)¹¹ for evaluation. Complete information about the DFC22 can be found in the [DFC22 site announcement](https://www.grss-ieee.org/community/technical-committees/2022-ieee-grss-data-fusion-contest/)¹².

10. <https://iee-dataport.org/competitions/data-fusion-contest-2022-dfc2022>

11. <https://codalab.lisn.upsaclay.fr/competitions/880>

12. <https://www.grss-ieee.org/community/technical-committees/2022-ieee-grss-data-fusion-contest/>

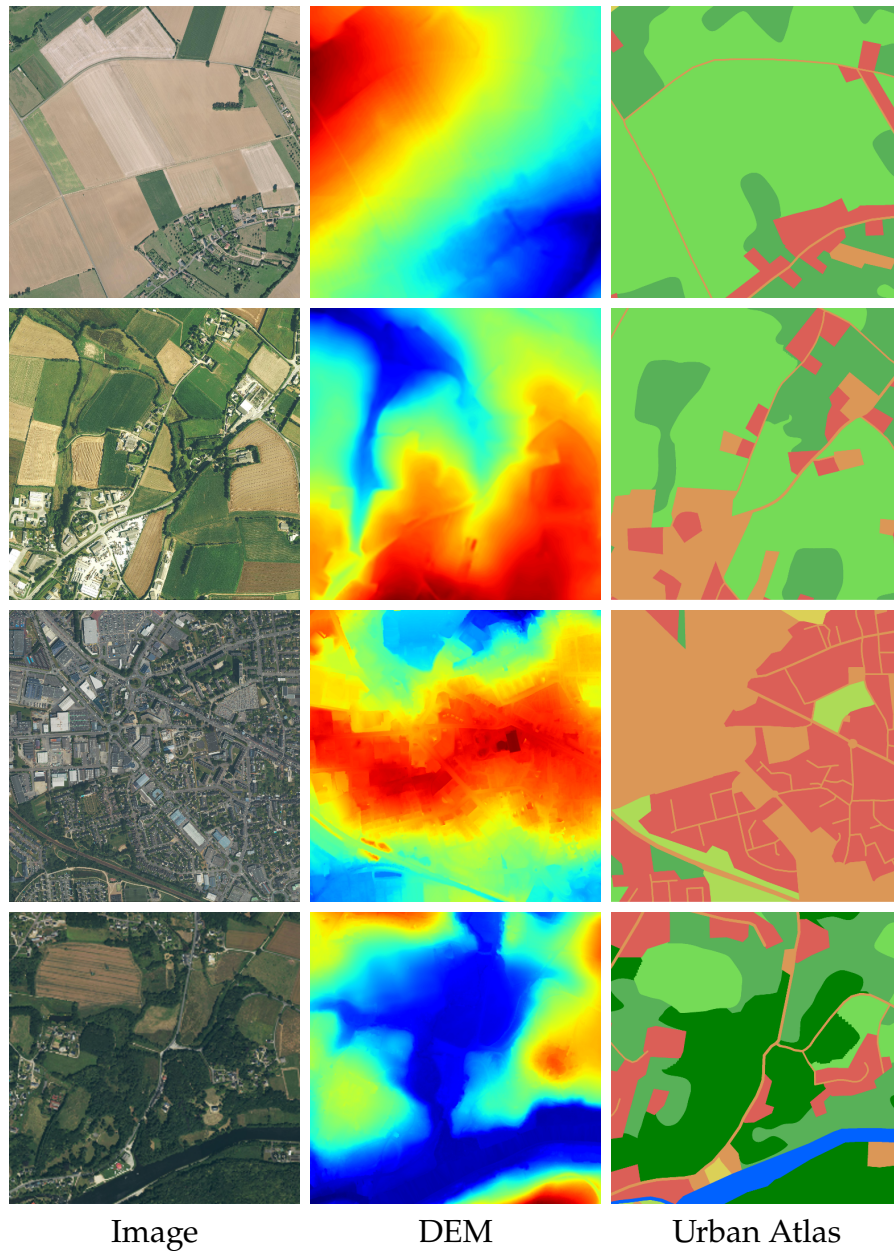


Figure 2.19 – Some samples of MF-DFC22 dataset on different locations in the training partition.

2.8 Conclusions

This chapter has presented an **analysis of existing Earth observation datasets from a critical point of view**: do they model real-life remote sensing applications? What do we expect from a *good* dataset? In general EO applications, one would like to train a model that generalize well across different geographic locations. Moreover, one typically has access to very few labeled data, while plenty of unlabeled data are available. Therefore, good EO datasets should recreate these situations to be considerate as a suitable and trustworthy evaluation benchmark.

Furthermore, in Section 2.2 we have investigated the learning capacities of **current supervised approaches on different settings**: on small-scale datasets, and at a large-scale multi-location set-up. Our experiments revealed that **common supervised semantic segmentation networks have generalization issues in the large-scale setting, if training data are not sufficiently varied**. Therefore, there is an opportunity for new learning paradigms to arise: semi-supervised learning, weakly-supervised learning, active learning, etc. In this work, we focus on semi-supervised learning techniques, because the plethora of unlabeled EO data available should be exploited to develop robust and generic models.

Having spotted the limitations of classic datasets and the opportunities of new training paradigms, Section 2.3 has introduced the **MiniFrance suite, a novel large-scale dataset designed for semi-supervised semantic segmentation in Earth observation**. MiniFrance has **unprecedented properties**, the diversity of landscapes and scenes reflects the complexity of reality. Above all, it was thoroughly designed for semi-supervised learning, including labeled and unlabeled data in its training partition and recreating a life-like application setting, which makes MiniFrance unique.

Moreover, Section 2.4 introduced two **tools that enable the analysis of a multi-location image dataset** in terms of representativeness between train and test sites: an appearance assessment tool (Section 2.4.1) and a class representativeness tool (Section 2.4.2). Appearance assessment is based on pre-trained CNNs to encode image appearance, t-SNE to reduce the dimension of the encoded appearance, and one-class SVM to estimate site distributions. Scores like IoU and IoT are then used to assess the similarity between data from different locations. Class representativeness is evaluated by comparing class distribution between different locations and by using the appearance assessment tool to understand the class distributions in the appearance space.

Section 2.5 presented a **comprehensive analysis of the MiniFrance data in terms of appearance similarity and class representativeness** by the means of the previously presented tools. This study has shown that MiniFrance is well-suited to address the semi-supervised learning problem.

We have also used the appearance assessment tool to demonstrate that MiniFrance is indeed more varied and complex than more classical single-location datasets such as the ISPRS Vaihingen. Thus, we hope that MiniFrance will contribute to push the research on the field to more challenging and realistic scenarios.

Finally, our work on **MiniFrance is part of the IEEE GRSS Data Fusion Contest 2022**. This year's contest is about semi-supervised learning for land cover classification. The MF-DFC22 dataset is based on MiniFrance, extending it to new modalities (DEM) and adding three new areas for evaluation. Thereby, we have contributed to the organization of this competition that will gather researchers from all over the world to find solutions to the semi-supervised problem.

SEMI-SUPERVISED LEARNING: DISCRIMINATIVE APPROACHES

Contents

Chapter summary	96
3.1 Introduction: discriminative models	97
3.2 Semi-supervised learning cast as multi-task	97
3.2.1 Multi-task learning	98
3.2.2 Multi-task semantic segmentation networks	100
3.2.3 Auxiliary tasks and losses	102
3.2.4 Experiments	106
3.3 Semi-supervised learning through consistency regularization	119
3.3.1 Vicinal risk minimization	121
3.3.2 FixMatch	123
3.3.3 Experiments	127
3.4 Conclusions	129

Chapter summary

This chapter explores semi-supervised learning from a discriminative perspective. In other words, we study methods that learn directly a function f that maps inputs x into their corresponding labels y (in a classification problem). Since we are interested in semi-supervised methods, these algorithms must be able to leverage unlabeled data to – somehow – refine the map function f . In this context, we investigate two groups of methods: multi-task learning strategies and consistency regularization-based approaches.

Multi-task learning was inspired by the human ability to use previous knowledge on related tasks to “better” learn a new task. In this framework, models are designed to learn several tasks simultaneously, keeping shared representations of the inputs among tasks. Thus, domain information contained in the training signals of one of the tasks can be seen as an inductive bias to other related tasks, which induces a regularization on the model and yields better generalization capacities.

In Section 3.2 we present different neural network architectures adapted to the semi-supervised multi-task semantic segmentation framework, as well as different auxiliary tasks and loss functions to apply together with the pixel-wise classification task. We perform experiments on three datasets suitable for semi-supervised learning that show the benefits of the proposed multi-task approach. However, we also observe the issues of multi-task learning: how to choose auxiliary tasks? how to choose the best-suited loss function? should we prefer parallel streams or sequential learning? early or late splitting?

The second part (Section 3.3) of the chapter is devoted to methods founded on the consistency regularization principle. Consistency regularization exploits the idea that semantically similar inputs should have similar predictions. This principle is widely applied in most of the current state-of-the-art semi-supervised classification approaches in computer vision. In particular, we delve into FixMatch, one of the most powerful semi-supervised classification methods to date, which is based on a bright combination of data augmentation, consistency regularization and pseudo-labeling. Our experiments on two public Earth observation benchmarks for scene classification show the high-performance of this model, even in extreme settings when very few labeled data are available during training. Moreover, they show how consistency regularization enhances the generalization capacities of the model with respect to new geographic locations, which we stated in Chapter 2 as an important feature in EO applications.

3.1 Introduction: discriminative models

Discriminative models comprise all the algorithms that model directly the posterior distribution $p(y|x)$. In other words, they learn directly a map function from inputs x to the outputs y (class labels in the classification problem or $y \in \mathbb{R}$ in a regression problem). The idea is to find the decision boundary between classes. On the other hand, generative models (which we will explore in Chapter 4) estimate the joint probability $p(x, y)$ of inputs and outputs, and then, by using the Bayes rule, they can compute $p(y|x)$ to get the prediction y .

Traditionally, there is an implicit collective consensus that discriminative classifiers are almost always preferred over generative ones. They usually yield superior performance on classification tasks, partly because they have fewer variables to compute. Ng et al. [156] have shown that, in general, generative models may converge faster, but discriminative models usually catch up and eventually surpass their performances. Another advantage of discriminative models over generative ones is that –since they are optimized for a specific task (estimate the posterior distribution)– prior knowledge can be introduced into the model.

Moreover, Vapnik [157] states in his *main principle of inference* that:

«(...) [one should] try to solve the problem directly and never solve a more general problem as an intermediate step. It is possible that the available information is sufficient for a direct solution but is insufficient for solving a more general intermediate problem.»

In this regard, we can say that generative models try to solve a general problem, namely, estimating the data distribution, as an intermediate step to determine $p(y|x)$. Instead, discriminative models directly estimate the decision boundary. Therefore, the latter should be preferred if our only goal is to find y . In this chapter we explore semi-supervised learning through two different discriminative approaches: first, with a multi-task learning approach and second, with a consistency regularization method. We apply these methods to Earth observation applications, in the form of semantic segmentation or scene classification.

3.2 Semi-supervised learning cast as multi-task

Are we, humans, multi-task mechanisms? Multi-tasking is the ability of executing several tasks at the same time. Even if sometimes we may think that we can simulta-

neously perform two tasks (that require a certain degree of concentration), this is pure illusion [158]. In reality, in any multi-task situation, when we have to execute two or more cognitive operations, at least one of the operations is slowed down. This is because our brain works in a similar way to a processor, switching tasks from one to another, not able to jointly perform them.

In spite of our inability to truly multi-task, what we call *multi-task learning* –in the context of machine learning– is inspired from human learning. Indeed, when we learn a *new* task, we are able to apply all the knowledge we have acquired by learning other tasks (i.e. our experience). Therefore, by **combining our knowledge** of several tasks, we can **improve our ability to learn** the new task [159]. This is the key idea of multi-task learning.

3.2.1 Multi-task learning

Multi-task learning (MTL) refers to machine learning algorithms that are designed to solve several problems simultaneously, in other words, algorithms that have multiple outputs (see Fig. 3.1). The objective is to improve the generalization capability of the model –on several tasks– by exploiting the common features and differences across tasks. We can achieve this by learning all tasks jointly, keeping a shared representation of the inputs. Thus, the information extracted to learn one task can be helpful to improve the knowledge about other tasks.

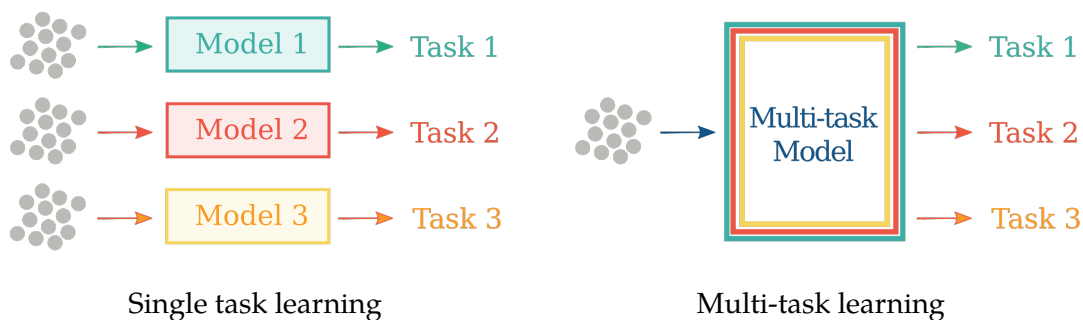


Figure 3.1 – Comparison between single task learning (left) and multi-task learning (right), example with three tasks. In STL each tasks needs to train a different model over the same inputs. In contrast, MTL needs just one model, sharing parameters, to perform all the tasks simultaneously.

Caruana [159] explains that the domain information contained in the training sig-

nals of one task can be seen as an inductive bias to other related tasks. Indeed, when part of the model is shared across tasks, that part of the model is more constrained, leading to an efficient and more generic feature extractor [30]. That is to say, MTL induces a regularization in the learning algorithm that results in better generalization and performance.

However, how all tasks do benefit from each other and how to make the best use of it is still unclear. Several studies [159–161] propose that for multi-task learning to be helpful, tasks must be related. The problem is then how to define related tasks, today’s choices being mostly intuitive or empirical: one can predict simultaneously characteristics of the road and the steering direction in a self-driving car [159]; or jointly predict phoneme duration and frequency profile in text-to-speech applications [162]. At the moment, there is no *theory of relatedness* to determine beforehand if a pair of tasks will help or hurt each other.

On the other hand, other studies show that we can benefit from unrelated tasks. For instance, using prior knowledge about unrelated tasks one can learn shared representations across tasks, and impose constraints on representations of unrelated tasks (such as orthogonality) [163]. However, we still need to know which tasks are related and which ones are unrelated.

Nevertheless, the general consensus is that MTL *can* improve the generalization capacities of the models, improve their performance or improve their convergence rate. Recent works on the subject show the benefits from multi-task learning at a large-scale, going up to training twenty-six tasks simultaneously [164], and showing that multi-task learning can reduce the need for labeled data [164, 165].

Based on all of the above, it is straightforward to imagine a multi-task learning model where some tasks are supervised (label-dependent) and others are completely unsupervised (no need for labels). This model would then be semi-supervised, because we can integrate completely unlabeled data into the learning process and benefit from the regularization induced by the auxiliary unsupervised tasks. A similar approach was proposed by Ando et al. [166], where unlabeled data are first leveraged in a MTL framework and then the model is fine-tuned on the primary supervised task.

Multi-task semi-supervised semantic segmentation

Inspired from this idea, the following sections explore semi-supervised multi-task learning for semantic segmentation. In this context, supervised semantic segmentation

is our primary task to solve and we study several unsupervised tasks to serve as auxiliary tasks, as well as neural network architectures adapted to this problem.

We aim to use unlabeled data to help generalization for semantic segmentation of aerial images. The challenge is two-fold: designing network architectures able to deal with both labeled and unlabeled images, and selecting unsupervised tasks to perform along with the appropriate auxiliary loss function.

Let $\phi_s(\cdot)$ be the function learned by a supervised segmentation network (for the sake of simplicity, the corresponding network will also be referred as ϕ_s). Such a network can be optimized through supervised learning using stochastic gradient descent and a classification loss \mathcal{L}_s (cross entropy loss is a standard choice). We denote x the input image and y the target label, then:

$$(x, y) \mapsto \mathcal{L}_s(\phi_s(x), y). \quad (3.1)$$

From a general point of view, using unlabeled data to help the previous optimization can be seen as a second task optimized with a loss function \mathcal{L}_u and a transfer function through the network denoted by ϕ_u . Without labels, unsupervised losses usually rely on comparing in some way the output to the input image:

$$x \mapsto \mathcal{L}_u(\phi_u(x), x). \quad (3.2)$$

In order to improve the genericity of ϕ_s , one has to relate ϕ_s and ϕ_u . This is generally done by partially sharing parameters between both networks. Finally, the semi-supervised loss is a weighted sum of the losses for each individual task:

$$\mathcal{L}(x) = \mathcal{L}_s(\phi_s(x), y) + \lambda \mathcal{L}_u(\phi_u(x), x). \quad (3.3)$$

3.2.2 Multi-task semantic segmentation networks

We propose here two types of semi-supervised networks which process the multi-task optimization –semantic segmentation as the supervised task, along with an unsupervised task– either as parallel streams or as sequential objectives (Figure 3.2).

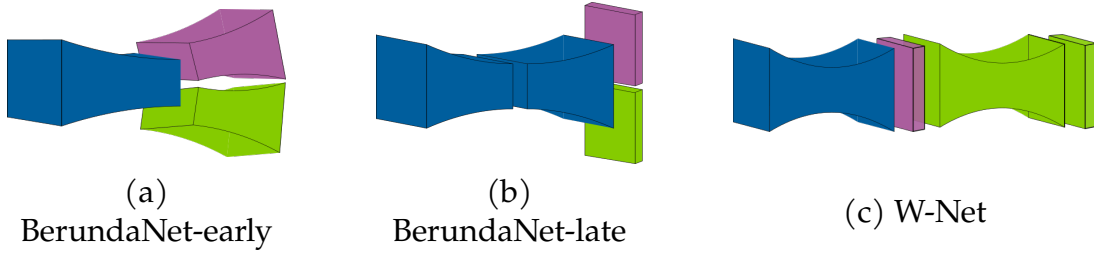


Figure 3.2 – Proposed multi-task neural network architectures for semi-supervised learning. Shared layers are depicted in blue, supervised layers are in purple, and unsupervised layers are shown in green.

BerundaNet (with early and late task splitting)

Standard encoder-decoder networks for semantic segmentation—such as SegNet [55] or U-Net [57]—can easily be extended for multiple task learning by adding a new head with a loss for the new, unsupervised task [92, 138]. With such an architecture (thereafter named BerundaNet after the mythological two-headed bird), both tasks have shared parameters until the data streams are split. We distinguish two variants depending on the splitting layer. Early splitting networks have one encoder and two decoders, one for each task (Fig. 3.2 (a)). On the contrary, with late-splitting task specialization occurs at the very end. It has an almost-all shared decoder with only a single separate convolutional layer for each task (Fig. 3.2 (b)).

Eventually, all architectures optimize the global loss defined in Eq. (3.3). \mathcal{L}_s can be any supervised loss for semantic segmentation, and in the following we consider the cross-entropy loss. \mathcal{L}_u is an unsupervised loss. In the experiments we will consider reconstruction losses (such as \mathcal{L}_1 or \mathcal{L}_2), unsupervised image segmentation losses and self-supervised losses that will be presented in Section 3.2.3.

W-Net [167, 168]

Multiple task learning can also be processed sequentially, as in W-Net [167] which combines two unsupervised objectives: segmentation and reconstruction. W-Net consists of two stacked U-Net [57], hence its name. We adapt the original design to semi-supervised learning by specializing the first U-Net block on the semantic segmentation task and focusing the second one on the unsupervised objective (Fig. 3.2 (c)). With respect to previous notations, in this case the network ϕ_s shares all parameters with ϕ_u . At the end of the first U-Net block, a soft-max layer is included to achieve the supervised

classification.

The loss function for our semi-supervised W-Net architecture is then more precisely decomposed as follows:

$$\mathcal{L}(x) = \mathcal{L}_s(\phi_s(x), y) + \lambda \mathcal{L}_u(\phi_u(\phi_s(x)), x), \quad (3.4)$$

where x is the input image, y its corresponding ground truth, $\phi_s(\cdot)$ represents the first U-Net block and $\phi_u(\cdot)$ represents the second U-Net block. As before, \mathcal{L}_s can be any supervised loss for semantic segmentation and \mathcal{L}_u is an unsupervised loss.

This kind of architectures –BerundaNet and W-Net– allows us to deal with both labeled and unlabeled data during training. When a labeled example is processed the gradient is backpropagated through the whole network, whereas if an unlabeled example is processed gradients are only backpropagated through the unsupervised part and shared parameters of the network (green and blue blocks in Figure 3.2). However, the main objective is still the semantic segmentation task. Thus, even if unsupervised parts are helpful during the training process, evaluation can be performed without them, which yields in standard-size inference networks.

3.2.3 Auxiliary tasks and losses

We now present some unsupervised losses \mathcal{L}_u which can leverage the information brought by images with no label. Two task objectives are usually considered, image reconstruction and image segmentation, leading to the following general formulation:

$$\mathcal{L}_u(\cdot) = \alpha^{(rec)} \mathcal{L}^{(rec)}(\cdot) + \alpha^{(reg)} \mathcal{L}^{(reg)}(\cdot), \quad (3.5)$$

where $\mathcal{L}^{(rec)}$ is a reconstruction loss, $\mathcal{L}^{(reg)}$ is a regularization loss and $\alpha^{(rec)}, \alpha^{(reg)}$ are balance coefficients.

In the following, we adapt some existing losses to semi-supervised semantic segmentation, and also propose a novel implementation of a *relaxed K-means* loss for unsupervised image segmentation. Moreover, we include in our study some self-supervised losses. Indeed, with the huge progress of self-supervision in representation learning [28, 29] they are an ineluctable topic of analysis for unsupervised learning.

Image reconstruction losses

Image reconstruction losses can be simply defined using solely standard reconstruction losses such as the classical \mathcal{L}_1 and \mathcal{L}_2 , as in equations (3.6) and (3.7). They enforce the encoding power of internal representations built by the network ϕ_s by closing the loop from it to the original input, the image itself. This kind of self-supervision is for example used in [167].

$$\mathcal{L}_1(x) = \frac{1}{N} \sum_{i=1}^N |x_i - \hat{x}_i|, \quad (3.6)$$

$$\mathcal{L}_2(x) = \frac{1}{N} \sum_{i=1}^N (x_i - \hat{x}_i)^2 \quad (3.7)$$

where x_i denotes the i^{th} pixel of the image, \hat{x}_i its reconstructed version and N the number of pixels in the image.

Image segmentation losses

Image segmentation aims to partition an image into multiple segments, where pixels in a segment share some properties, like color, intensity, or texture. This task can be performed in an unsupervised manner –based on the input image only– and might be a better complement to the supervised semantic segmentation task. We consider in this work two different unsupervised losses to perform unsupervised image segmentation.

Relaxed K-means. We propose a new loss for unsupervised image segmentation, which combines the old intuitions behind the k -means algorithm with the expressive power of neural network’s non-linear modeling. In a standard manner, it is cast as a color image quantization problem, where the objective is to find an optimal, reduced set of K colors for encoding the image. Formally, it minimizes the reconstruction loss $\mathcal{L}^{(rec)}(x, x_c)$ where x_c is the quantized image.

We still denote x the input image and x_i its value at pixel i . k -means alternatively optimizes centroids of color clusters c_k ($k \in \{1, K\}$) and membership matrices $\hat{y}^{(k)}$ of x to cluster k . It follows:

$$c_k = \frac{\sum_i x_i \hat{y}_i^{(k)}}{\sum_i \hat{y}_i^{(k)}} \quad (3.8)$$

and

$$x_c = \sum_{k=1}^K c_k \cdot \hat{y}^{(k)}. \quad (3.9)$$

In standard k -means, memberships $\hat{y}_i^{(k)} \in \{0, 1\}$ are then determined such that $\|x_i - c_k\|^2$ is minimum. Instead, we relax the hard constraint so that $\hat{y}_i^{(k)} \in [0, 1]$ and estimate memberships as the output $\hat{y} = \phi(x)$ of a network which minimizes $\mathcal{L}^{(rec)}(x, x_c)$. In our experiments we will use:

$$\mathcal{L}_{km}^{(rec)}(x) = \mathcal{L}_1(x, x_c). \quad (3.10)$$

Eventually, to compensate for the relaxation we add a regularization term which ensures memberships are peaked to a one-cluster-per-pixel distribution:

$$\mathcal{L}_{km}^{(reg)}(x) = \sum_{k=1}^K \sum_i \hat{y}_i^{(k)} \cdot (1 - \hat{y}_i^{(k)}). \quad (3.11)$$

The whole unsupervised loss is then in the form of Eq. (3.5).

Mumford-Shah Loss. Recent works on unsupervised image segmentation have brought the power of level set methods based on minimization of the Mumford-Shah functional [169] in CNNs [170].

The unsupervised segmentation loss is then expressed as:

$$\mathcal{L}_{MS}(x) = \sum_{k=1}^K \sum_i |x_i - c_k|^2 \hat{y}_i^{(k)} + \alpha^{(reg)} \sum_{k=1}^K \sum_i |\nabla \hat{y}_i^{(k)}|, \quad (3.12)$$

where we kept the same notations as before.

In Eq. (3.12), the first term corresponds to the reconstruction loss, while the regularization term penalizes gradient variations in the resulting segmentation, thus leading to more homogeneous regions.

Self-supervised losses

Recently, self-supervised methods have shown impressive results on learning data representations. The main idea behind self-supervision is to build a supervised task from completely unlabeled data by producing labels from the data themselves. Many self-supervised tasks have been proposed lately, and we explore here two pretext tasks

to perform along with semantic segmentation, that can be easily integrated to our semi-supervised framework.

Inpainting. Similarly to the context autoencoder [171], we aim to solve the problem of filling in a missing piece in the image. The loss function is then expressed in terms of L_2 distance as

$$\mathcal{L}_{ca}(x) = \mathcal{L}_2(M \odot x, M \odot \phi_u((1 - M) \odot x)), \quad (3.13)$$

where M is a binary mask (value of 1 for dropped pixels and 0 for input pixels) and \odot the element-wise product.

There is an intrinsic hyperparameter to the inpainting problem: c , the crop size to mask from the image. In our experiments we try $c \in \{80, 160\}$ and in Section 3.2.4 we report results for $c = 80$ since it led to the best results. In our settings, masks are randomly chosen over the image.

Jigsaw puzzle. Solving jigsaw puzzles using neural networks was first proposed by [172] to learn visual representations. In brief, the task consists in cutting out the image into 9 patches, shuffle them and train the network to retrieve the original image.

In practice, we follow here a similar approach to [173], where a network is trained to solve two tasks simultaneously (in our case, the jigsaw puzzle and the semantic segmentation) and the input is an image with permuted patches. The problem is then formulated as a classification task, using standard cross-entropy loss. We use the maximal Hamming distance algorithm from [172] to define a set of P allowed patch permutations. In our experiments we compare results for $P \in \{30, 100\}$. Since $P = 100$ led to the best results, we report them in Section 3.2.4.

3.2.4 Experiments

In this section, we evaluate the multi-task semi-supervised semantic segmentation framework presented above, with two main objectives: first, to confirm the value of datasets as tinyMiniFrance and MiniFrance to the semi-supervised problem; and second, to prove that leveraging unlabeled data into the learning process can yield better generalization and improves the model’s performance.

To this end, we perform experiments on three semantic segmentation datasets suitable for the semi-supervised settings: tinyMiniFrance, MiniFrance (see Chapter 2) and Christchurch (CASD) [136, 137]. Since tinyMiniFrance was especially designed for fast development and validation times, we perform a thorough analysis of our multi-task learning framework on this dataset, including a comparison of neural network architectures and auxiliary losses. Experiments on MiniFrance built upon the results obtained on the tiny version of the dataset, and are more succinct, due to long computing times. Finally, Christchurch experiments extend the analysis made on tinyMiniFrance to self-supervised losses and hyper-parameters tuning.

Experiments on tinyMiniFrance

The purpose of this section is to show that we can benefit from semi-supervised learning –using unlabeled data during the learning process– to achieve better results and generalization than vanilla supervised approaches.

To this end, we perform experiments to compare a semi-supervised setting with an equivalent supervised approach, using different backbone architectures. First, we train supervised networks (SegNet and U-Net) in a classical way, using the cross-entropy loss, over the labeled training split of tinyMiniFrance. Secondly, we train a BerundaNet-late architecture (with SegNet and U-Net backbone) over tinyMiniFrance –using both, labeled and unlabeled data–, which is the equivalent semi-supervised strategy. We train BerundaNet-late with a reconstruction task (\mathcal{L}_1 as auxiliary loss) and with an unsupervised segmentation task (\mathcal{L}_{km} as auxiliary loss) and show that in both cases, semi-supervised learning can improve the results obtained by the supervised network.

Results of these experiments are summarized in Table 3.1. The *oracle* corresponds to the hypothetical case where annotations are available for all training cities (i.e, we can access the ground-truth for all the images of the 8 regions in the training split) during the training phase. The oracle results might be seen as an upper bound for semi-

supervised learning strategies and they are brought out here just for comparison and not as a result of this work.

Table 3.1 – Supervised vs. Semi-supervised experiments over tinyMiniFrance using different backbone architectures. We refer to the hypothetical case where annotations are available for all 8 training regions as *oracle*. Semi-supervised denotes results for BerundaNet-late with the corresponding backbone.

Backbone	Oracle		Supervised		Semi-supervised (BerundaNet-late)			
	\mathcal{L}_{ce}		\mathcal{L}_{ce}		$\mathcal{L}_{ce} + \lambda\mathcal{L}_1$		$\mathcal{L}_{ce} + \lambda\mathcal{L}_{km}$	
	OA	mIoU	OA	mIoU	OA	mIoU	OA	mIoU
SegNet	59.06	23.95	36.76	14.03	45.52	14.43	42.26	15.75
U-Net	57.71	25.25	46.30	18.18	47.90	18.70	46.92	18.26

Along with Table 3.1, Figure 3.3 shows segmentation maps obtained during the testing phase for the previous experiments with a SegNet backbone. We refer as *undisclosed* to the entries that are not publicly available but that are shown here as a reference and comparison to our results: ground-truth and oracle. At a global scale, we observe that semi-supervised methods –whether with reconstruction or with segmentation auxiliary task– present more homogeneous and finer segmentation maps than their supervised counterpart. This is noticeable in particular in clear roads and less noisy regions. Adding unlabeled data during the learning process helps to regularize and generalize better, especially in the case of MiniFrance data, where labels are often approximate. In some cases, semi-supervised methods can even beat the oracle predictions, as in the last row example where the oracle mistook a pasture section for a water section.

Several remarks can be raised from these results:

- First, MiniFrance is challenging. The oracle shows that even if we could access all images labels (of the 8 cities in the training split) during training, we would only get 59% overall accuracy with a fully supervised approach (see Table 3.1, oracle column). This is far below the accuracy that can be achieved with other datasets.
- The amount of labeled data influences a lot the performance of supervised settings. Focusing on the results of the oracle and the supervised experiment (second and third columns on Table 3.1), we see that for a SegNet architecture going from 8 to 2 training labeled cities implies a 22% loss in accuracy and 10% less of mIoU.

And even if the U-Net seems more robust to the amount of labeled data, reducing annotated data diminishes network performances notoriously. From a visual perspective, prediction quality is noticeably worse for the supervised approach with respect to the oracle (third and fourth columns in Figure 3.3).

- Semi-supervised strategies exhibit promising results. In both cases, whether we use a SegNet or a U-Net backbone, the benefits of semi-supervised learning are clear, regardless of the chosen auxiliary task there is a gain of accuracy and mIoU with respect to the supervised method.
- Finally, from a visual perspective, semi-supervised methods (fifth and sixth columns in Figure 3.3) are superior to the supervised one (fourth column). Indeed, semi-supervised segmentation maps are more homogeneous than the supervised ones (see the second, fourth and sixth row examples). Besides, urban cartography is better delineated in the semi-supervised semantic maps and seems more appropriated with respect to the original image.

Those are encouraging results for future works on semi-supervised learning for semantic segmentation.

► Influence of the choice of architecture on semi-supervision

In the following, we compare the architectures presented in section 3.2.2 with respect to both auxiliary tasks, reconstruction (using \mathcal{L}_1 loss) and unsupervised segmentation (with \mathcal{L}_{km} loss). For the BerundaNet-early architecture a SegNet backbone is used. Results of these experiments are reported in Table 3.2.

Table 3.2 – Neural networks for semi-supervised semantic segmentation comparison.

<i>Auxiliary Loss</i>	<i>Architecture</i>	<i>Backbone</i>	<i>OA (%)</i>	<i>mIoU (%)</i>
\mathcal{L}_1	BerundaNet-early	SegNet	35.94	9.51
	BerundaNet-late	SegNet	45.52	14.43
	BerundaNet-late	U-Net	47.90	18.70
	W-Net [167]	U-Net	40.72	13.79
\mathcal{L}_{km}	BerundaNet-early	SegNet	38.20	10.26
	BerundaNet-late	SegNet	42.26	15.75
	BerundaNet-late	U-Net	46.92	18.26
	W-Net [167]	U-Net	45.20	16.13

Whatever the chosen auxiliary task, BerundaNet-late with U-Net backbone is the architecture that achieves the best scores, followed by W-Net and BerundaNet-late with SegNet backbone. BerundaNet-early is just slightly better than a supervised approach with same backbone. This indicates that, in terms of network architecture, it might be better to split the supervised and unsupervised tasks rather late, enabling more shared parameters. Thus, the image statistics learned through optimization of the auxiliary task are better harnessed for the main objective.

Figures 3.4 and 3.5 show some examples of semantic maps and unsupervised outputs at inference time for these methods, using reconstruction and unsupervised segmentation as auxiliary task, respectively. From these examples, we confirm that whether we choose reconstruction or segmentation as auxiliary unsupervised task, BerundaNet-late (U-Net backbone) gets the finer and smoother results, especially in the second case.

Therefore, the choice of the architecture and backbone matters for the semi-supervised task. BerundaNet-late performs better than BerundaNet-early with same backbone. Moreover, the U-Net backbone outperforms the SegNet backbone. Finally, the simple architecture BerundaNet-late presented in this work places it first, before W-Net.

Thus, it seems the choice of architecture is at least as important as the loss design. This choice does not only rely on the number of parameters (W-Net has about twice the number of parameters of BerundaNet, since it relies on two U-Nets) but also how the supervised and unsupervised information are mixed.

► Influence of the choice of auxiliary loss on semi-supervision

In this section, we analyze the effect on the semantic segmentation results of different auxiliary losses presented in section 3.2.3. To this end, we train the same network architecture while changing the loss. We choose BerundaNet-late with U-Net backbone, since it was the network with the best scores in the previous sections, regardless of the auxiliary task.

Table 3.3 reports the results obtained through these experiments. Figure 3.6 exhibits some examples of segmentation maps and unsupervised outputs obtained by BerundaNet-late with reconstruction losses (\mathcal{L}_1 and \mathcal{L}_2) at inference time, while Figure 3.7 shows examples using unsupervised segmentation as auxiliary task.

For the reconstruction task, \mathcal{L}_1 loss outperforms the \mathcal{L}_2 approach, this is confirmed by visual examples in Fig. 3.6 where we perceive that results are marginally better for \mathcal{L}_1 than for \mathcal{L}_2 in terms of smoothness, especially in urban areas like the third and fourth

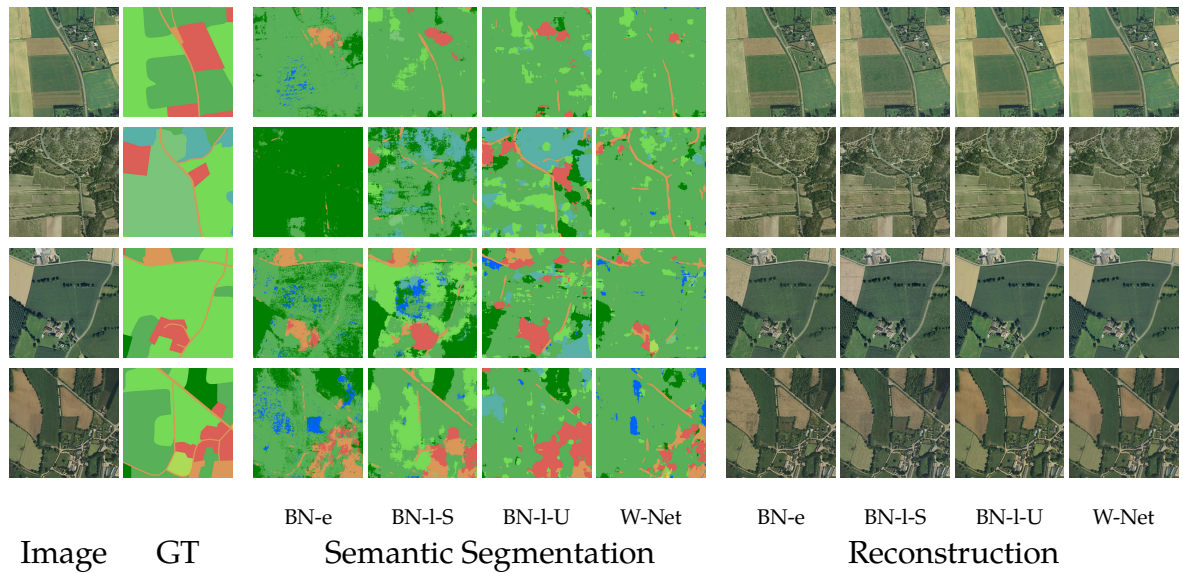


Figure 3.4 – Results comparison for different neural network architectures with reconstruction as auxiliary task (\mathcal{L}_1 auxiliary loss). BN-e stands for BerundaNet-early, BN-I-S/BN-I-U for BerundaNet-late with SegNet/U-Net backbone, respectively.

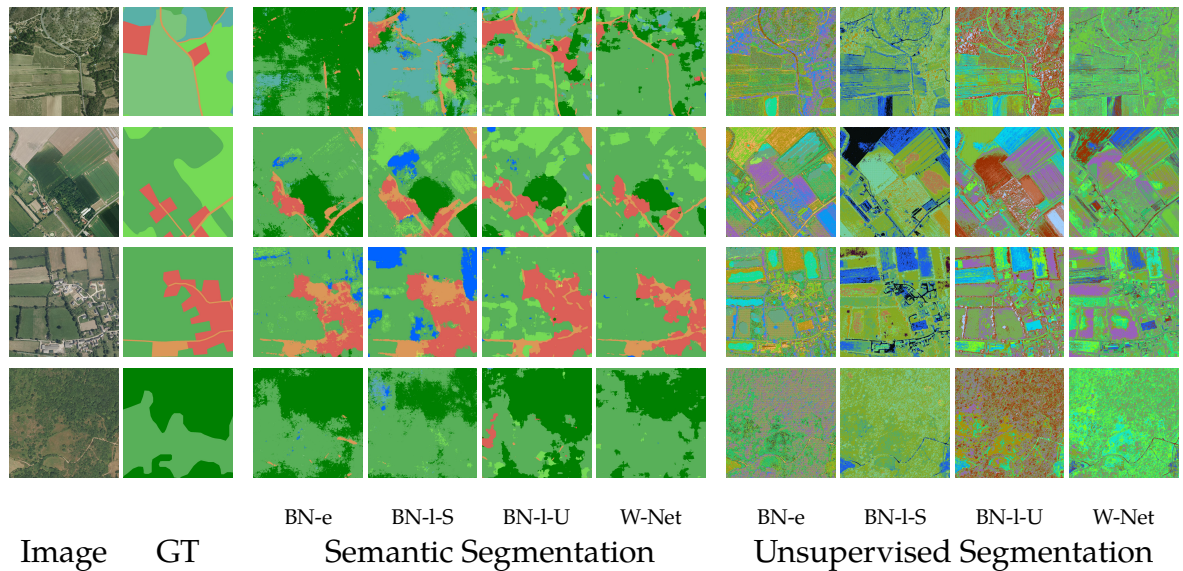


Figure 3.5 – Results comparison for different neural networks with unsupervised segmentation as auxiliary task (\mathcal{L}_{km} auxiliary loss). BN-e stands for BerundaNet-early, BN-I-S/BN-I-U for BerundaNet-late with SegNet/U-Net backbone, respectively.

row examples.

In the case of segmentation, \mathcal{L}_{km} and \mathcal{L}_{MS} are somehow equivalent. However, from Figure 3.7 the \mathcal{L}_{km} loss seems to be superior to \mathcal{L}_{MS} in most cases, especially when it comes to road detection.

Table 3.3 – Auxiliary unsupervised loss effect comparison using BerundaNet-late with U-Net backbone.

<i>Auxiliary Task</i>	<i>Aux. Loss</i>	<i>OA (%)</i>	<i>mIoU (%)</i>
Reconstruction	\mathcal{L}_1	47.90	18.70
	\mathcal{L}_2	44.55	16.27
Segmentation	\mathcal{L}_{km}	46.92	18.26
	\mathcal{L}_{MS} [170]	46.88	18.57

Experiments on MiniFrance

All the results and analysis exposed above were conducted using the tinyMiniFrance dataset, due to computing capacity and processing time. In this section we present the first semi-supervised results over the entire MiniFrance dataset.

To this end, we train a BerundaNet-late with U-Net backbone as it is the best result we got in a semi-supervised setting (see Table 3.1). We use our regularized k-means loss (\mathcal{L}_{km}) as auxiliary unsupervised loss. We also train a U-Net network on the labeled partition of MiniFrance in a classic supervised way for comparison with the semi-supervised setting. Results are reported in Table 3.4 and some visual results of the semi-supervised experiment are shown in Figure 3.8.

These results on MiniFrance are coherent with previous ones reported with tinyMiniFrance. They confirm our hypothesis that tinyMiniFrance is a good representation of the entire MiniFrance dataset. Moreover, they confirm that including unlabeled data during the learning process helps to improve the results on semantic segmentation.

It is worth to mention that training these models over the entire MiniFrance dataset for 450 pseudo-epochs takes roughly 3 weeks. While inference time –processing all the tiles on the testing partition– takes about 6 days (with a single GPU).

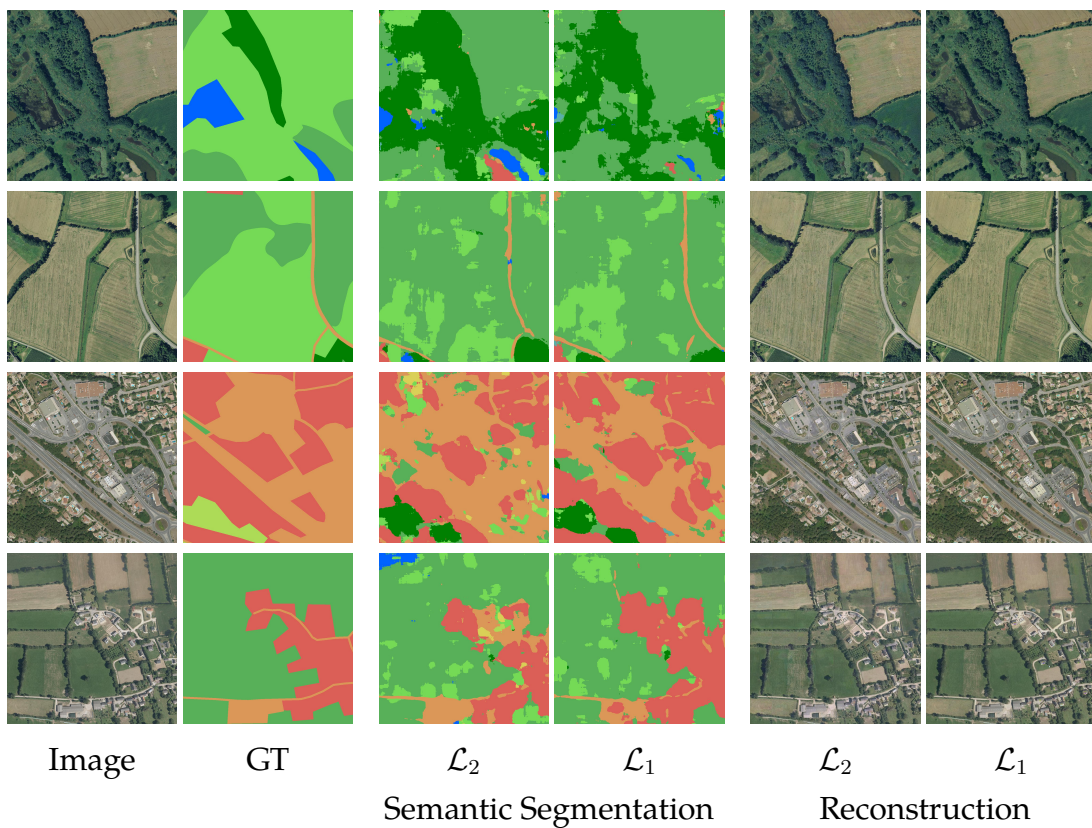


Figure 3.6 – Segmentation maps and reconstruction outputs for BerundaNet-late (U-Net backbone), using different unsupervised reconstruction losses for the auxiliary task.

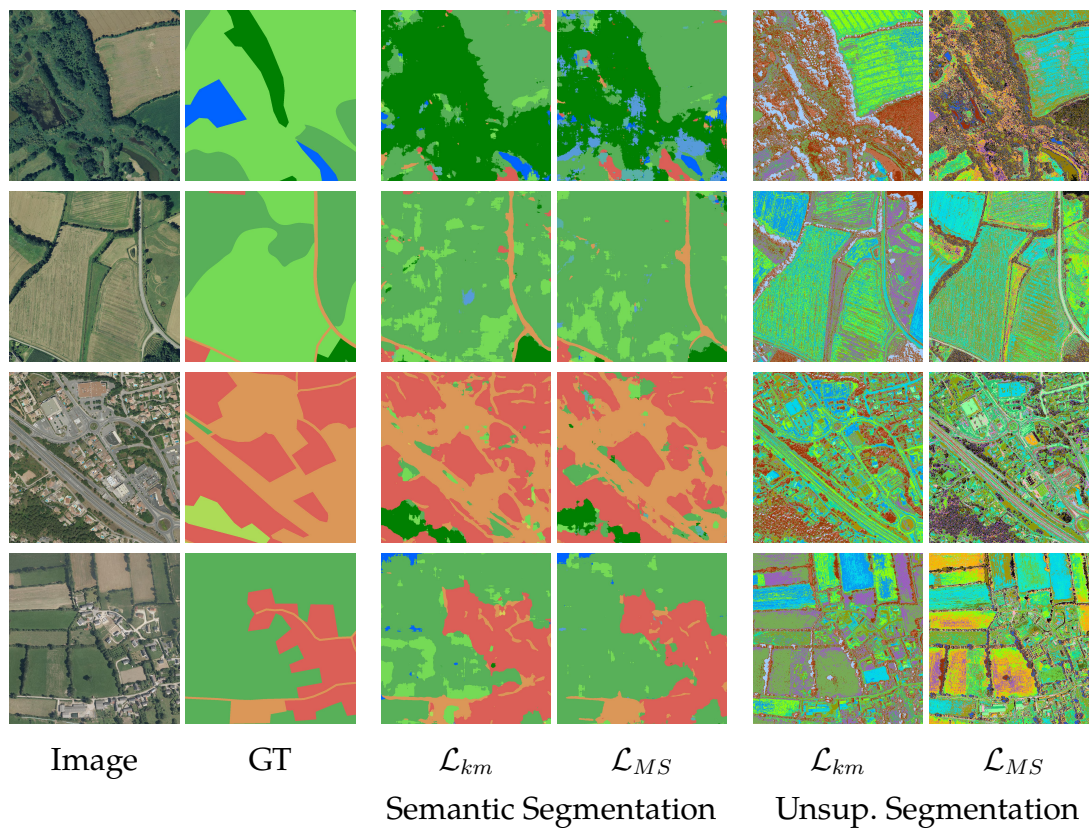


Figure 3.7 – Semantic segmentation maps and unsupervised segmentation outputs for BerundaNet-late (U-Net backbone), using different unsupervised segmentation losses for the auxiliary task.

Table 3.4 – First semi-supervised results over MiniFrance.

<i>Method</i>	<i>Network</i>	<i>Backbone</i>	<i>Aux. Loss</i>	<i>OA</i>	<i>mIoU</i>
Supervised	U-Net	U-Net	-	44.28	20.77
Semi-Supervised	BerundaNet-late	U-Net	\mathcal{L}_{km}	45.16	21.20

Experiments on Christchurch

We also perform experiments on the Christchurch Aerial Semantic Dataset (CASD) ¹ to test the reliability of our framework.

CASD comprises aerial imagery at 10 cm/px resolution over Christchurch, New Zealand. Dense semantic annotations were produced by ONERA/DTIS on 4 images, considering 4 classes: buildings, cars, vegetation and background [136, 137]. The dataset also includes 20 aerial images without annotations, which makes it suitable for semi-supervised learning algorithms.

For these experiments, we use a training partition containing labeled and unlabeled data –2 annotated tiles and 20 non-annotated tiles–, and keep 2 annotated tiles for validation. We train a BerundaNet-late architecture with U-Net backbone, because of its simplicity and efficiency. The network is trained during 50 pseudo-epochs with 5000 labeled iterations and 5000 unlabeled iterations. Since the dataset allows it (training only takes a few hours), we also evaluate different values of the hyperparameter λ (in Equation (3.3)).

Results are reported in Table 3.5. Mean and variance are obtained over 4 runs of each experiment. We note that semi-supervised methods outperform the supervised setting. Moreover, best scores are obtained with unsupervised segmentation losses, and especially our relaxed K-means loss which improves the mIoU score by +3.39% and overall accuracy by +1.97%, with respect to the supervised setting.

Figure 3.9 shows two examples of segmentation maps obtained by the different methods. In the first row example, the supervised approach is the only one that mistakes the river as a building; the supplementary information provided by unlabeled images to the semi-supervised methods allows us to prevent this error. In the second

1. Available at <https://doi.org/10.5281/zenodo.3566005>



Figure 3.8 – Semi-supervised results over MiniFrance. BerundaNet-late with U-Net backbone and \mathcal{L}_{km} as auxiliary loss.

Table 3.5 – Results comparison for supervised and semi-supervised methods over the Christchurch Aerial Semantic Dataset.

<i>Mode</i>	<i>Aux. Task</i>	<i>Aux. Loss</i>	λ	<i>OA (%)</i>	<i>mIoU (%)</i>
Sup	-	-	-	81.06 \pm 0.46	67.43 \pm 0.49
Semi-sup	Rec	\mathcal{L}_1	0.5	82.28 \pm 0.55	68.78 \pm 1.27
		\mathcal{L}_2	5	82.36 \pm 0.42	68.99 \pm 0.85
	Seg	\mathcal{L}_{km}	1	83.03 \pm 0.42	70.82 \pm 0.35
		\mathcal{L}_{MS}	1	82.94 \pm 0.26	70.24 \pm 0.84
	Self	\mathcal{L}_{ca}	5	82.57 \pm 0.59	69.47 \pm 0.7
		\mathcal{L}_{js}	0.5	82.88 \pm 0.95	70.17 \pm 1.12

row, the \mathcal{L}_{km} loss is the only one that correctly segments the central building, likely due to its color clustering capacity.

In general, we observe from the experiments over CASD that including unlabeled data during training helps to improve the segmentation maps with respect to the case where we only use our limited labeled data.

Influence of the λ Hyperparameter. We also study the impact of the weighting parameter λ on the segmentation performance. Figure 3.10 illustrates the average behavior of each loss with respect to the value of λ .

Three behavioral groups appear. Segmentation losses are robust to the choice of λ and show, in general, better performances. \mathcal{L}_1 and \mathcal{L}_{js} work better for small λ and require cautious hyperparameter tuning, as they are close to the fully-supervised case. \mathcal{L}_2 and \mathcal{L}_{ca} losses show the same optimum for $\lambda = 5$, which comes likely from the fact that inpainting uses \mathcal{L}_2 to estimate discrepancies.

In a multi-task setting, different tasks might have very different behaviors and orders of magnitude. Tuning a weighting hyperparameter is not straightforward and further work is needed to find a neat normalization. Some works have even focused on adapting the multi-task loss balancing during training [174].

This section has shown the benefits of integrating unlabeled data into the learning process through a multi-task learning perspective. Our experiments show that, indeed, semi-supervised learning is helpful to improve generalization. However, a multi-task

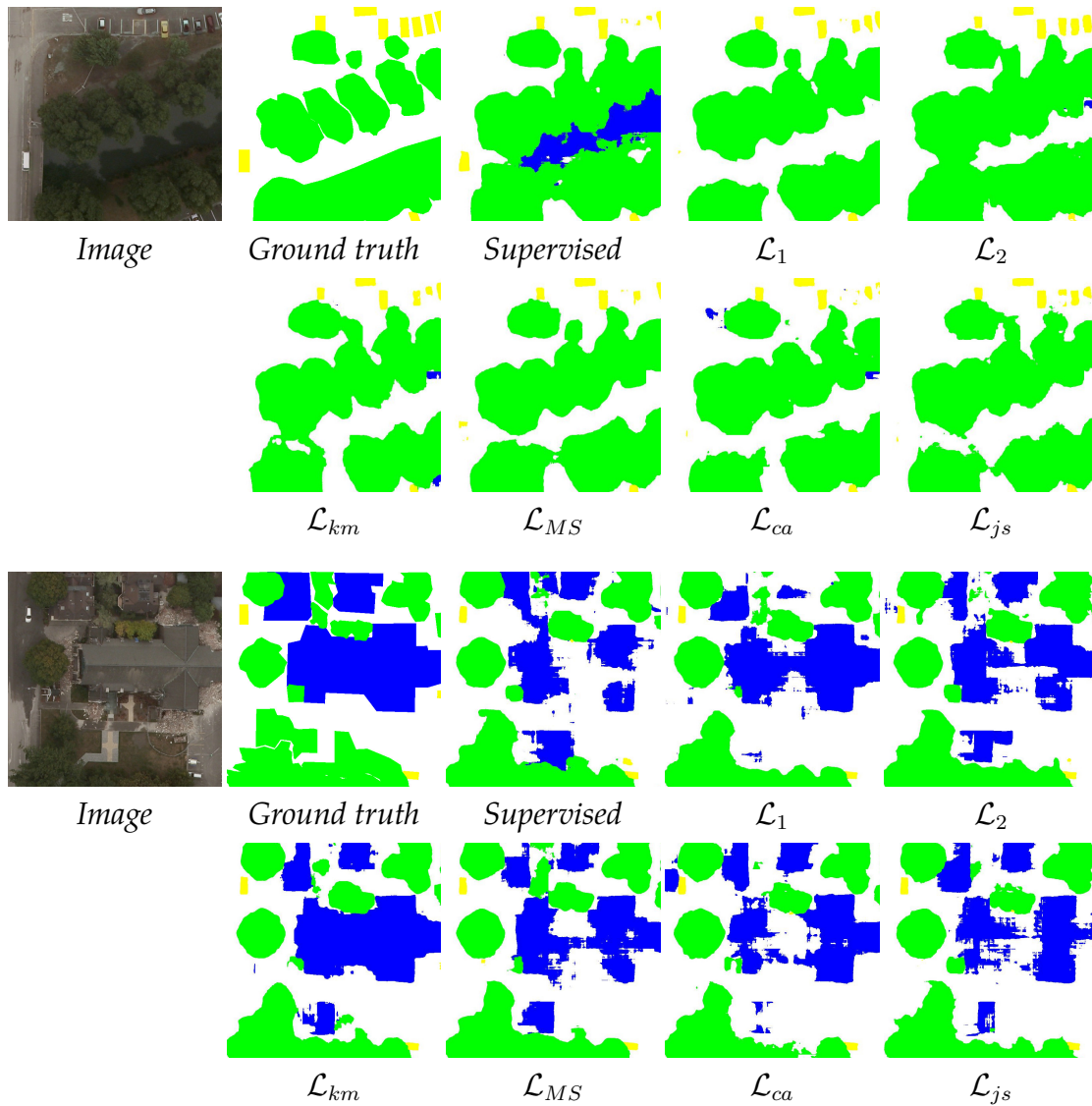


Figure 3.9 – Two examples of inference over the CASD dataset. ■ buildings, ■ cars, ■ vegetation and □ background.

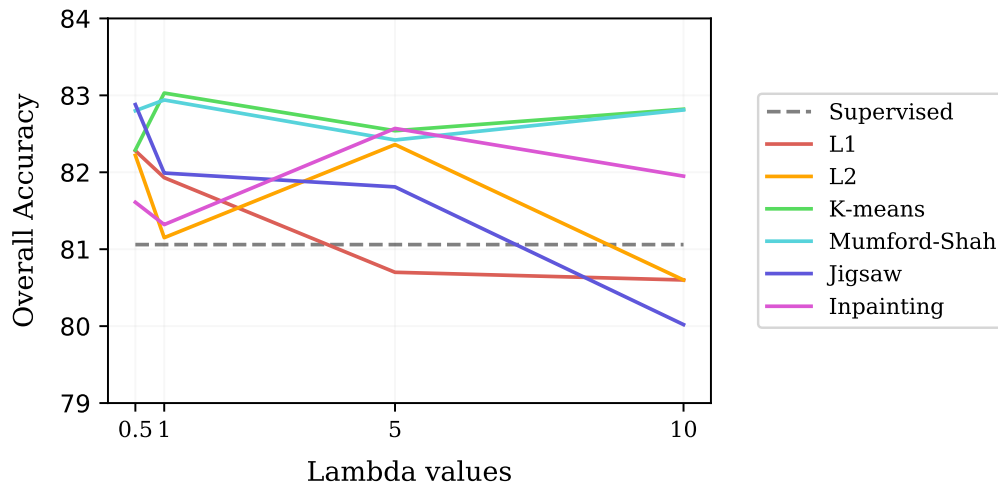


Figure 3.10 – Impact of the λ parameter on the semantic segmentation performance.

approach has its drawbacks. It involves decisions about the architecture (parallel or sequential streams?), parameter sharing (early or late splitting?), auxiliary tasks to use, hyper-parameters to tune, etc. In contrast, new emerging strategies leverage more decisive mechanisms. Next section is dedicated to one of the most broadly used approach in current computer vision algorithms: consistency regularization.

3.3 Semi-supervised learning through consistency regularization

Dehaene [158] says: « our brain is much more flexible [than current machines]. It quickly manages to prioritize information and, whenever possible, extract general, logical, and explicit principles. »

In other words, the human brain is capable of creating abstract representations of our world. Our abstraction capabilities give us the ability to interpolate, extrapolate or even create/imagine new versions of objects that we know. For instance, we may have never seen a real panda bear in our lives, we may have seen pictures or drawings of pandas –only 2D representation of these giant animals–, however, the day we visit the zoo and see a real panda, we will recognize it immediately. This is because of the abstract representation we have created from the pictures and drawings, because we are able to extrapolate them to the 3D world. Our brain is able to create an abstract

representation of the panda, and thus different images (pictures, drawings or the live bear) activate similar representation signals in our heads.

This is the idea behind *consistency regularization*. First introduced by Bachman et al. [175], consistency regularization enforces the idea that realistic perturbations of data points should not significantly change the output of the predictor. More precisely, if a

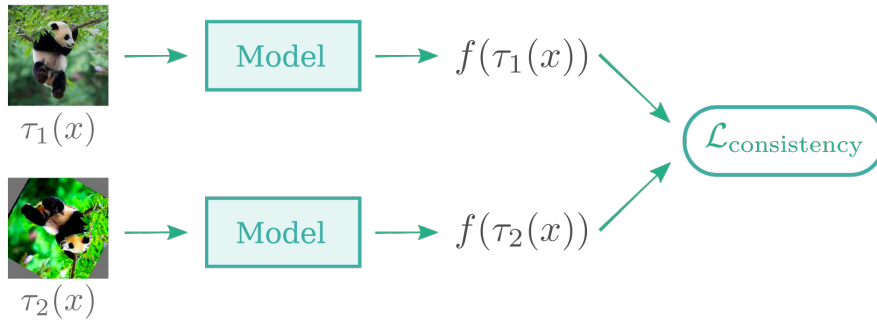


Figure 3.11 – Consistency regularization by comparing the outputs of two transformations of the same image.

model is fed with semantically similar inputs, the output of the model should also be similar. Fig. 3.11 illustrates the consistency regularization principle. Usually, this regularization is imposed as a loss term that can be written as:

$$\mathcal{L}_{\text{consistency}} = \frac{1}{N} \sum_{i=1}^N \ell(f(\tau_1(x_i)), f(\tau_2(x_i))), \quad (3.14)$$

where $\{x_i\}_{i=1}^N$ are data samples, ℓ is a function that ensures proximity between its inputs (usually a cross-entropy term or a L_2 norm), and τ_1, τ_2 are random perturbations applied to the inputs (usually sampled from a fixed set of transformations \mathcal{T}).

Semi-supervised methods in deep learning developed to date exploit, in general, two principles: the first one is consistency regularization, that we just defined [64–66, 176]; and the second one is pseudo-labeling [67]. Pseudo-labeling is an iterative process where a model is initially trained on the labeled data, the pre-trained model is used to make predictions on the unlabeled data, then one can select the unlabeled samples where the model was confident of its prediction and consider them as pseudo-labeled examples to expand the labeled training set; and repeat.

Current state-of-the-art semi-supervised methods for classification, such as MixMatch [71] or FixMatch [73], usually combine these ideas, achieving impressive results.

However, they rely heavily on data augmentation, which works well on the image domain, but can be hard to adapt to other use-cases.

In this section we explore consistency regularization-based methods. First, we propose a theoretical context based on vicinal risk minimization to motivate and justify the consistency loss; then we recall the principles of FixMatch, the current state-of-the-art method for semi-supervised classification in computer vision; and finally, we study whether this method is suitable for Earth observation applications on scene classification.

3.3.1 Vicinal risk minimization

In a supervised learning problem, we want to find a function $f \in \mathcal{F}$ that maps feature vectors into a corresponding target vector, $f : x \in \mathcal{X} \mapsto y \in \mathcal{Y}$. Where $(x, y) \sim P$. Our goal is to find a predictor f such that its error with respect P is as small as possible. To this end, we define $\ell : \mathcal{Y} \times \mathcal{Y} \mapsto \mathbb{R}$, a loss function that penalizes the differences between the prediction $f(x)$ and the real target y . Then, we minimize the expected risk:

$$\mathcal{R}[f] = \mathbb{E}_P[\ell(f(X), Y)] = \int_{\mathcal{X} \times \mathcal{Y}} \ell(f(x), y) dP(x, y). \quad (3.15)$$

In practice, the joint distribution P is usually unknown. However, we have access to a set of data $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$, where $(x_i, y_i) \sim P, \quad \forall i \in \{1, \dots, N\}$. Our goal is then to find the predictor f such that its error with respect to these data is as small as possible.

Data \mathcal{D} can be used to approximate the risk \mathcal{R} by the means of the empirical distribution:

$$d\hat{P}_\delta(x, y) = \frac{1}{N} \sum_{i=1}^N \delta_{x_i}(x) \delta_{y_i}(y), \quad (3.16)$$

where δ_x is the Dirac mass function centered in x .

Using this empirical distribution defined by equation (3.16), we can estimate the expected risk by the empirical risk:

$$\hat{\mathcal{R}}[f] = \frac{1}{N} \sum_{i=1}^N \ell(f(x_i), y_i). \quad (3.17)$$

Solving the learning problem through estimating f by minimizing the empirical risk defined in the equation above, equation (3.17), is known as *Empirical Risk Minimization* (ERM) [157].

However, the $d\hat{P}_\delta$ estimate is just one way of estimating the distribution P and one can consider improved estimates of the distribution. For instance, Chapelle et al. [177] propose to replace δ_{x_i} in equation (3.16) by an estimate of the density in the vicinity of x_i , $dP_{x_i}(x)$. This gives an empirical distribution of the form:

$$d\hat{P}_{\text{vic}}(x, y) = \frac{1}{N} \sum_{i=1}^N dP_{x_i}(x) \delta_{y_i}(y). \quad (3.18)$$

Estimating f by minimizing the empirical risk defined by $d\hat{P}_{\text{vic}}$ is known as *Vicinal Risk Minimization* (VRM). VRM assumes that a sample point shares the same label with other samples in its vicinity. This is exactly the idea behind –and that justifies– the use of data augmentation in our learning algorithms.

Zhang et al. [178] propose to extend the vicinal risk minimization by approximating the distribution P in a vicinity of the pair (x_i, y_i) (instead of x_i alone), $\nu_{(x_i, y_i)}(\cdot, \cdot)$, which defines an empirical distribution \hat{P}_ν as:

$$d\hat{P}_\nu(x, y) = \frac{1}{N} \sum_{i=1}^N \nu_{(x_i, y_i)}(x, y). \quad (3.19)$$

Following this idea, we can express the vicinity $\nu_{(x_i, y_i)}$ based on consistency regularization as:

$$\nu_{(x_i, y_i)}((x_1, y_1), (x_2, y_2)) = dP_{x_i}(x_1) dP_{x_i}(x_2) \delta_{y_i}(y_1) \delta_{y_i}(y_2). \quad (3.20)$$

Therefore, the vicinal risk induced by consistency regularization can be expressed as:

$$\hat{\mathcal{R}}_{\text{consistency}}[f] = \frac{1}{N} \sum_{i=1}^N \int \ell(f(x_1), f(x_2)) dP_{x_i}(x_1) dP_{x_i}(x_2) \delta_{y_i}(y_1) \delta_{y_i}(y_2), \quad (3.21)$$

which makes the expression of the risk based on consistency, in Equation (3.21), interesting is the fact that it does not need labels y to be computed. Indeed, the loss term $\ell(f(x_1), f(x_2))$ does not depend on y , and the values of y_1 and y_2 depend exclusively on the chosen sample (x_i, y_i) . However, the value of y_i is irrelevant for computing the risk. This property is particularly appealing on applications where labels may not be available, namely semi-supervised or unsupervised applications.

Therefore, the consistency regularization loss term in Equation (3.14) is directly connected to a vicinal risk minimization problem, which allows us to understand its effect-

tiveness and makes it an interesting term to explore in semi-supervised applications.

3.3.2 FixMatch

FixMatch [73] is a method recently proposed by the Google Brain team that surpasses –with quite simple ideas– the state-of-the-art approaches to semi-supervised classification. The model is mainly based on the two principles that we mentioned above: pseudo-labeling and consistency regularization.

Notation. Let $\mathcal{X}_\ell = \{(x_i, y_i)\}_{i=1}^N$ be a batch of N labeled samples, with x_i the training samples (typically images) and y_i their corresponding labels. Let p_i denote the one-hot version of y_i . Let $\mathcal{X}_u = \{u_i\}_{i=1}^{\mu N}$ be a batch of μN unlabeled samples, with μ being an hyper-parameter defining the relative size of \mathcal{X}_u with respect to \mathcal{X}_ℓ . Let $p_m(y|x)$ be the predicted class distribution given by the model for input x . $H(p, q)$ denotes the cross-entropy of distribution q relative to distribution p . Finally, strong augmentations are noted $\mathcal{A}(\cdot)$, while weak augmentations are noted $\alpha(\cdot)$.

Pseudo-labeling[67]. Also known as self-training, pseudo-labeling is based on the idea of using the model itself to generate artificial labels for unlabeled data. The labels generated this way are known as *pseudo-labels*.

Usually, the model is first trained on the available labeled data. Then, it is used to make predictions on the unlabeled data. Unlabeled samples satisfying certain properties (typically, high confidence predictions) are selected to expand the training dataset and re-train the model over this new (pseudo-)labeled data. More precisely, pseudo-labeling retains the hard labels predicted by the model (i.e. the argmax of the model’s output) of the unlabeled data whose predictions are confident, this is, whose largest class probability fall above a certain threshold. Let $q_i = p_m(y|u_i)$, the pseudo-labeling loss function is usually defined as:

$$\mathcal{L}_{\text{pseudo-label}} = \frac{1}{N} \sum_{i=1}^N \mathbb{1}\{\max(q_i) \geq \omega\} H(\hat{q}_i, q_i), \quad (3.22)$$

where $\hat{q}_i = \text{arg max}\{q_i\}$ and ω is a predefined threshold for the prediction confidence.

Pseudo-labeling is closely related to entropy regularization [179], where the model’s predictions are encouraged to present low-entropy.

Consistency regularization. FixMatch uses unlabeled data to compute a regularization term based on the premise that the model should output similar predictions when fed different transformed versions of the same sample. In the FixMatch framework, the consistency regularization loss –whose general form can be found in equation (3.14)– is written as:

$$\mathcal{L}_{\text{consistency}} = \sum_{i=1}^N \|p_m(y|\tau_1(u_i)) - p_m(y|\tau_2(u_i))\|_2^2, \quad (3.23)$$

where τ_1, τ_2 are random transformations applied to u_i , sampled from a fixed set of transformations \mathcal{T} .

One of the main novelties of FixMatch –and probably the key to its success– is the combination of strong and weak transformations to apply the consistency regularization term.

Data augmentation. A key point of the success of FixMatch is the bright use of data augmentation to perform consistency regularization. The method distinguishes two kinds of data augmentation techniques:

- ▶ Weak augmentations. They include vertical and horizontal flips as well as vertical and horizontal translations.
- ▶ Strong augmentations. The original work explores two families of strong augmentations: RandAugment [180] and CTAugment [71]. For simplicity, in this work we perform our experiments considering RandAugment as strong augmentation strategy. Let \mathcal{T} be a set of transformations (rotations, translations, brightness perturbations, etc), RandAugment is controlled by two parameters N and M . When RandAugment is applied to a data sample, it chooses N transformations among the set \mathcal{T} to be applied sequentially. For each of these N transformations, the magnitude of the severity of the distortion is sampled on the range defined by M .

Loss function. As in many semi-supervised algorithms, the FixMatch loss function consists in two loss terms: a supervised loss term \mathcal{L}_s , which is applied to labeled data only, and an unsupervised loss \mathcal{L}_u which is applied to only images (no need for labels).

- ▶ The supervised loss term \mathcal{L}_s is just a standard cross-entropy loss applied to weakly augmented labeled samples:

$$\mathcal{L}_s = \frac{1}{N} \sum_{i=1}^N H(y_i, p_m(y|\alpha(x_i))). \quad (3.24)$$

- The unsupervised loss term is a combination of pseudo-labeling, consistency regularization and different types of data augmentation. For each unlabeled sample², FixMatch generates a pseudo-label, using weakly-augmented versions of each image. This is, it computes the predicted class distribution $q_i = p_m(y|\alpha(u_i))$ and uses \hat{q}_i as a pseudo-label (if the confidence of the model is larger than a certain threshold ω). Then, it enforces the similarity of the model's prediction of a strongly-augmented version of the same image, through the cross-entropy loss:

$$\mathcal{L}_u = \frac{1}{\mu N} \sum_{i=1}^{\mu N} \mathbb{1}\{\max(q_i) \geq \omega\} H(\hat{q}_i, p_m(y|\mathcal{A}(u_i))). \quad (3.25)$$

- The final loss is simply a weighted sum of both terms exposed above:

$$\mathcal{L}_{\text{FixMatch}} = \mathcal{L}_s + \lambda_u \mathcal{L}_u, \quad (3.26)$$

where λ_u is an hyper-parameter of the model.

Hyper-parameters. At first glance, FixMatch seems to be a very straightforward and simple method. However, behind the simplicity and cleverness of the combination of pseudo-labeling and consistency regularization, there are other elements that contribute significantly to the success of the method: hyper-parameters choices and other training strategies (optimizer, training schedule, etc). As the authors point out in their work, some important factors of the training framework are weight decay regularization, the optimizer choice (SGD with momentum) and the learning rate decay choice (cosine weight decay). In our experiments we also observe that the use of batch normalization layers is essential for the convergence of the method.

In this work, we follow the same hyper-parameters setting as in the original paper, using an unofficial Pytorch implementation³.

Fig. 3.12 illustrates the FixMatch training pipeline.

2. In practice, labeled samples are included in the set of unlabeled samples, without their labels.

3. <https://github.com/LeeDoYup/FixMatch-pytorch>

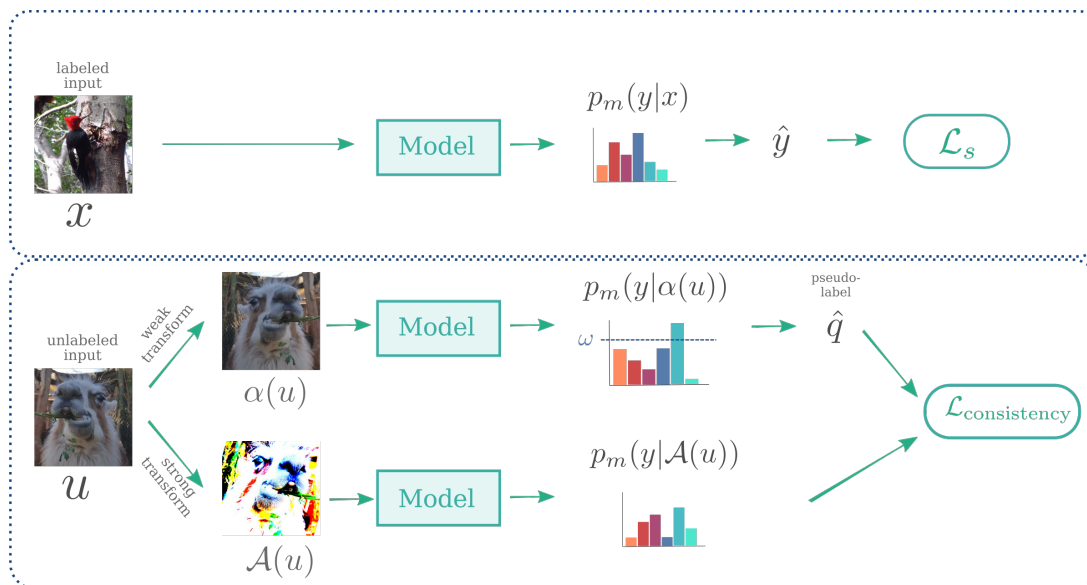


Figure 3.12 – Fixmatch overview. In a nutshell, if a labeled sample x is given to the model it follows the pipeline on top, which is a standard supervised classification setting. If an unlabeled sample u is processed, the bottom pipeline is applied: a weakly transformed version of the input $\alpha(u)$ and a strongly transformed version $\mathcal{A}(u)$ both pass through the model. From the prediction of $\alpha(u)$ one gets a pseudo-label \hat{q} if the confidence of the prediction is above a threshold ω , this pseudo-label is used to compute a consistency loss term together with the prediction of $\mathcal{A}(u)$.

3.3.3 Experiments

We perform experiments with this appealing framework on two publicly available EO datasets for scene classification: EuroSAT [124] and So2Sat LCZ42 [125]. The goal is to assess whether this kind of models can be applied to EO data as successfully as in the vision domain.

The **EuroSAT Dataset** comprises patches from Sentinel-2 images over 34 countries in Europe. Each patch is labeled with one of 10 land cover/land use classes (e.g. industrial, residential, highway, pasture, forest, etc.). Classes are well-balanced, with 2,000 to 3,000 examples per class, 80% of which are used for training. We use the EuroSAT RGB version.

The **So2Sat LCZ42 Dataset** is composed of Sentinel-1 and Sentinel-2 image patches over 42 locations over the globe. Patches are labeled according to the Local Climate Zones scheme (LCZ), with 17 categories. It is worth to mention that the training set and testing set are geographically independent, containing images from different locations. This makes this dataset particularly difficult, because models need to be sufficiently robust to generalize well on the test data. For our experiments, we only make use of the RGB Sentinel-2 bands (B04, B03, B02), as in EuroSAT.

Experimental settings. To assess the semi-supervised classification capacities of FixMatch on these datasets, we train this framework on a small subset of labeled examples and the entire dataset as unlabeled data.

We vary the number of labeled samples per class on which the model is trained, and compare our results with a fully supervised baseline, Wide-ResNet, which is the backbone of the FixMatch algorithm. Wide-ResNet, as a supervised method, is trained on labeled data only, while FixMatch is trained on the whole dataset, using labels when available.

Results. Table 3.6 summarizes the obtained results. As we may observe, FixMatch is undeniable superior to the supervised Wide-ResNet baseline in all settings of this experiment, including the fully supervised setting where all labels of the training data are considered and both models are then trained on the same amount of data.

As expected, we observe that the gap between Wide-ResNet and FixMatch scores decreases as the number of labeled examples available for training increases. However, even in the fully supervised case this gap cannot be neglected, since it represents 1.2%

Dataset	Labeled samples/class	% of labels	Wide-ResNet	FixMatch [73]
	2000 on avg.	100%	97.56 \pm 0.52	98.81 \pm 0.06
EuroSAT	100	\sim 5%	86.36 \pm 0.26	97.83 \pm 0.12
	20	\sim 1%	62.93 \pm 1.01	95.78 \pm 0.99
	10	\sim 0.5%	52.33 \pm 1.59	94.95 \pm 1.12
	5	\sim 0.25%	43.83 \pm 3.18	94.45 \pm 1.29
	1	\sim 0.05%	28.02 \pm 0.97	67.46 \pm 4.67
	\sim 20000	100%	50.93 \pm 0.16	63.00 \pm 0.49
So2Sat	1000	\sim 5%	44.17 \pm 0.40	61.68 \pm 0.53
	200	\sim 1%	35.45 \pm 0.17	60.41 \pm 0.66
	100	\sim 0.5%	30.90 \pm 0.35	56.51 \pm 0.68

Table 3.6 – Classification results using FixMatch (Accuracy [%] \uparrow). Comparison with a purely supervised method (Wide-ResNet) on two scene classification benchmarks, EuroSAT and So2Sat LCZ42. Best scores in bold.

of accuracy on EuroSAT and 12% on So2Sat.

In the case of the EuroSAT dataset, FixMatch is particularly robust, with a performance over 94% of accuracy when trained with very few labels per class (5, 10 or 20 labeled samples per class). Its accuracy drops to 67% when it is given only one labeled example per class, but still remains far superior to Wide-ResNet in the same labeled-data regime.

On the So2Sat dataset, results are more modest, with results varying from 56% to 63% of accuracy. However, this was expected since training and test images are geographically independent, inducing a domain adaptation problem. The performance gap is even more pronounced with respect to the supervised Wide-ResNet in the fully supervised setting with a difference of 12% and FixMatch is still far superior when very few labeled data are available, with a difference of 26 points of accuracy when trained on only 100 labeled samples per class.

These results are encouraging and show the interest of methods of this kind for Earth observation applications. Consistency regularization, with the adequate data transformations, allows us to add robustness to domain gaps into our models. This is a very appealing and desirable property in EO, as we discussed in Chapter 2.

3.4 Conclusions

This chapter was devoted to the study of **semi-supervised learning from a discriminative perspective**. In this context, we have studied two families of algorithms: **multi-task learning methods and consistency regularization-based approaches**.

In the multi-task setting, we have introduced **deep multi-task neural networks** to perform semi-supervised semantic segmentation. In particular, **we presented BerundaNet** –a simple extension of classic encoder-decoder architectures– which proves to be very effective in the semi-supervised task, and that works better than existing architectures, like W-Net. Together with these architectures, **we explored unsupervised auxiliary losses to use alongside with semantic segmentation**. Especially, we introduced the **relaxed k-means loss** to perform unsupervised image segmentation.

Our experiments on three publicly available benchmarks for semi-supervised semantic segmentation have shown that **we can benefit from unlabeled data during the learning process to improve semantic segmentation maps**. Indeed, semi-supervised approaches provide finer and more homogeneous predictions. We also observed that a simple architecture like BerundaNet-late with a suitable backbone such as U-Net is enough to enhance the segmentation performances.

Nevertheless, the problem of semi-supervised learning is not solved yet. We have seen that these multi-task approaches *can* improve semantic segmentation results, but it is not always the case. In a multi-task approach as the ones presented in this chapter, one must be careful on the choice of architecture and the auxiliary task to perform along. Furthermore, there exist other possible ways to solve the semi-supervised problem. For instance, one could develop generative models to learn the intrinsic distribution of data from labeled and unlabeled examples and use this information together with labels to improve the segmentation. Another possibility is the use of pseudo-label methods that propagate labels from annotated examples through non-annotated ones, based on a confidence criterion, to enlarge available training data.

The second part of the chapter explores methods based on the consistency regularization principle. **Consistency regularization** is one of the most widely applied techniques in current semi-supervised classification algorithms. It enforces the idea that a model should output similar predictions for semantically similar examples. We have proposed a **theoretical framework based on vicinal risk minimization** to justify the use of consistency regularization. Then, we have experimentally assessed **FixMatch**, the

current state-of-the-art method for semi-supervised classification in computer vision, on two publicly available Earth observation datasets for scene classification.

Our experiments demonstrate the **effectiveness and transferability of methods such as FixMatch to the Earth observation domain**. Moreover, they show that consistency regularization, with the right set of data transformations, **enhances the robustness of the models with respect to domain shifts**, which is a desirable feature in EO applications. The main contribution of the second part of this chapter is then the successful application of these new kinds of models to EO data.

SEMI-SUPERVISED LEARNING: GENERATIVE APPROACHES

Contents

Chapter summary	132
4.1 Introduction: generative models	133
4.2 Energy-based models	138
4.2.1 Joint energy-based models (JEM)	140
4.2.2 Semi-supervised learning with JEM	142
4.3 Experiments	143
4.3.1 Joint classification and generation with JEM	145
4.3.2 Semi-supervised classification with JEM	147
4.3.3 Out-of-distribution analysis	150
4.3.4 Application to land cover mapping	152
4.3.5 Can we combine FixMatch and JEM?	155
4.4 Limitations	156
4.5 Perspectives: semantic segmentation with JEM	157
4.6 Conclusions	159

Chapter summary

This chapter studies semi-supervised learning with generative models. Unlike discriminative methods –that we described in Chapter 3– generative models aim to get a deep understanding of the data, to capture their natural features, by learning the data distribution $p(x)$ and not only the posterior distribution $p(y|x)$.

We first introduce in Section 4.1 the main existing deep generative frameworks, describing their fundamentals. In particular, we delve into energy-based models (EBMs) (see Section 4.2) because their simplicity allows to integrate label information into the generative model in a very natural way. Indeed, EBMs are generative models that capture all the information about inputs through a scalar energy function, which defines a data distribution. Since there is no constraint on the function to estimate the energy, EBMs possess a matchless flexibility with respect to other generative frameworks. Furthermore, by the means of neural networks, we can model very complex energy functions, with impressive results in several appealing applications.

Taking this into account, this chapter focuses on a recent framework for generative modeling and explore its applicability to Earth observation images. The joint energy-based model (JEM, see Section 4.2.1) learns an EBM to estimate the joint distribution of the data and the categories, $p(x, y)$, obtaining a neural network that is able to classify and synthesize images. Moreover, it extends naturally to a semi-supervised setting. Indeed, since the loss function can be decomposed in two terms –one label-dependent and a second one that only depends on samples alone (no label needed)– one can easily integrate unlabeled samples into the training loop.

In Section 4.3, we perform experiments on various public Earth observation datasets for scene classification, and we show that JEM performs on par with state-of-the-art methods on scene classification and image generation. Furthermore, in semi-supervised experiments, JEM outperforms standard classifiers when very few labeled data are available for training. Finally, we show that models of this kind allow us to address high-potential applications such as out-of-distribution analysis and land cover mapping with confidence estimation.

This chapter concludes with a theoretical extension of JEM to semantic segmentation in Section 4.5. However, since this extension is still at early stages, practical validation is needed and left for future works.

4.1 Introduction: generative models

We have previously explored semi-supervised methods from a discriminative perspective, learning models that estimate the conditional distribution $p(y|x)$ (see Chapter 3). **Generative models** follow a different –and, in a certain way, more general– approach. Their goal is to model the intrinsic properties of data, by estimating the distribution $p(x)$, from which data samples are drawn. In particular, **generative classifiers** model the joint distribution of data and labels $p(x, y)$, then by the means of the Bayes rule (Eq. (4.1)), they compute $p(y|x)$ to output a prediction y :

$$p(y|x) = \frac{p(x, y)}{p(x)}. \quad (4.1)$$

As it was said in Chapter 3, in a classification problem –where the only goal is to find a function f that maps inputs to labels, $x \mapsto y$ – it is usually preferred to use discriminative classifiers because they have been designed to directly find decision boundaries between classes, instead of solving a more complex problem as the assessment of the data distribution $p(x)$, or the joint distribution $p(x, y)$ [157, 181, 182]. In the context of **semi-supervised learning**, both unlabeled samples from $p(x)$ and labeled samples from $p(x, y)$ are available and are used to estimate $p(y|x)$. Hence, *new* questions naturally arise: since we have abundant unlabeled data, could we use them to get a better estimation of $p(x)$? At the same time, can this better estimation of $p(x)$ help us to find a better approximation of $p(y|x)$?

Indeed, the goal of a classification framework (either supervised or semi-supervised) is to learn a model that yields similar representations for samples coming from the same class. Therefore, knowing (or approximating) the distribution $p(x)$ can provide particularly useful information about how to group similar examples in a representation space [30]. Then, one can conceive frameworks where a generative model of $p(x)$ shares parameters with a discriminative model of $p(y|x)$, similarly to a multi-task setting, inducing prior information into the supervised classification task [183]. For instance, Salakhutdinov and Hinton [184] have shown that methods of this kind, that use unlabeled data for modeling $p(x)$ (in particular, a deep belief network), can greatly improve the estimation of $p(y|x)$.

Deep generative models

Deep generative models [185] comprise a family of techniques which aim to learn the intrinsic data distribution by the means of neural networks. Their ultimate goal is essentially to get a deeper understanding of data, by learning automatically the natural features of a dataset, its categories or dimensions. They are also useful for many real-life applications like super-resolution, image denoising, inpainting or neural network pre-training.

Even if when we refer to generative models, one may automatically think of generative adversarial models (GANs), because of their undeniable success on image generation, current research on deep generative models includes a wide variety of approaches which can be grouped in different categories. They may differ on the objective to optimize, the motivations and the optimization process. In the following we briefly describe the main ideas behind each group.

Generative Adversarial Networks (GANs) [186]. They have been a breakthrough on image generation since their appearance back in 2016. GANs are composed of two networks, a discriminator and a generator. The discriminator D estimates the probability that a sample x comes from the data distribution $x \sim p_d(x)$. Meanwhile, the generator G generates new data from a latent space, $z \sim p_z(z)$, and tries to capture the data distribution p_d by fooling the discriminator into thinking its generated samples are real. Today, StyleGAN2 [187] is the state-of-the-art GAN model, improving quality of generated images and training performance. Moreover, GANs have been used to tackle semi-supervised classification: first introduced by Salimans et al. [188], the SGAN modified the discriminator architecture to achieve classification and discrimination of generated samples, with good results on semi-supervised classification.

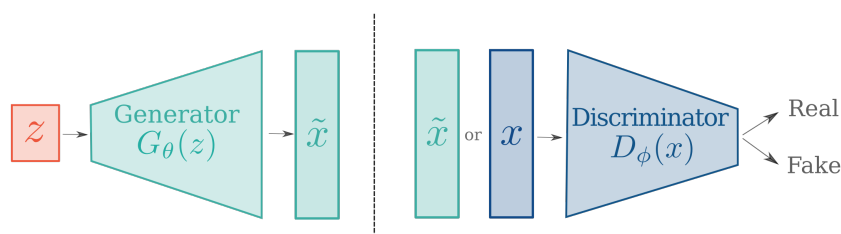


Figure 4.1 – Generative adversarial network (GAN). Two networks –a generator (G_θ) and a discriminator (D_ϕ)– are simultaneously trained with an adversarial optimization.

Variational Autoencoders (VAEs) [189]. Also well-known and widely used generative models, VAEs have two components in their architecture too: an encoder E , and a decoder D . The goal of E is to encode the inputs into a latent space \mathcal{Z} . The decoder, similarly to the generator in a GAN, takes a vector $z \in \mathcal{Z}$ and outputs an image. The difference from a traditional autoencoder comes from the fact that instead of mapping directly input x into a single point $z \in \mathcal{Z}$, the encoder E maps x onto a distribution $p_z(z|x)$ from which we can sample z . Moreover, unlike GANs, whose latent space distributions are predefined, VAEs learn the latent space distributions during training. As of 2020, NVAE [190] –a carefully designed hierarchical architecture– was the first successful VAE model applied to natural images as large as 256×256 pixels. Variational Autoencoders have also been used for semi-supervised classification. Indeed, back in 2014, Kingma et al. [68] described different models to achieve semi-supervised classification based on VAEs, exploiting the information contained in the data density using generative models.

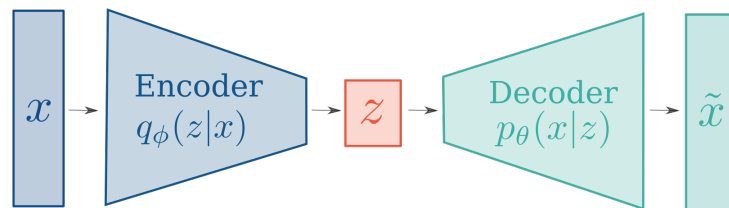


Figure 4.2 – Variational autoencoder (VAE). Two networks –an encoder (q_ϕ) and a decoder (p_θ)– are simultaneously trained by using variational inference.

Diffusion Models [191]. Inspired by non-equilibrium thermodynamics, diffusion models have surprised with their capacity to generate realistic images and, at the same time, keep a tractable likelihood function. In a nutshell, they define a Markov chain of diffusion steps to slowly add random noise to data as $q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1 - \beta_t}x_{t-1}, \beta_t I)$, where β_t is an hyperparameter and I is the identity matrix. Then a neural network is trained to reverse the diffusion process and generate data samples from noise. Recently, diffusion models have shown impressive performances, with results that beat GANs in image generation [192]. Another remarkable application of these novel diffusion models is text-conditional image synthesis [193]. To our knowledge, diffusion models have not been exploited to achieve semi-supervised classification yet.

Normalizing flows [194]. Normalizing flows aim to map simple distributions (den-

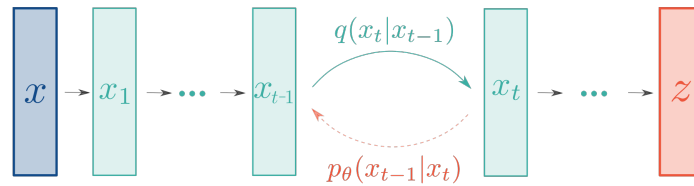


Figure 4.3 – Diffusion models. They go from x in the input space to z in the latent space by iteratively adding Gaussian noise to the previous step. This defines a Markov chain $q(x_t|x_{t-1})$. Then, a neural network (p_θ) is trained to reverse the diffusion process and generate samples from noise.

sities that are easy to sample and evaluate) to complex ones (data distribution). To this end, they make use of the change of variable formula. This formula describes how to evaluate densities of a random variable that is a deterministic function of another random variable. In a nutshell, they train a neural network composed of differentiable and invertible layers to transform data $x \sim p_d(x)$ into noise $z \sim p_z(z)$, where $p_z(z)$ is a simple distribution. Then, since this map is reversible, one can sample from $p_z(z)$ and obtain \hat{x} in the input space. In 2018, Glow [195] overcame existing generative normalizing flows and showed that flows using invertible 1×1 convolutions can efficiently synthesize realistic-looking, large images, with competitive results on standard benchmarks. More recently, Pumarola et al. [196] proposed a conditioning scheme for normalizing flows that enables compelling applications for multi-modal data modeling, which includes image manipulation, style transfer and multi-modal mapping. Normalizing flows have been applied to semi-supervised classification. For instance, Izmailov et al. [197] proposed FlowGMM, which uses a latent Gaussian mixture model to tackle semi-supervised learning with normalizing flows in a wide range of data domains.

Energy-based Models (EBMs) [198, 199]. Inspired by statistical physics, energy-based models are probably the first deep generative models that have been explored, since Boltzmann machines, restricted Boltzmann machines or deep belief networks. However, due to long sampling times, they were left behind for a while. This family of models captures dependencies between variables only through a scalar function, known as the *energy function*. Therefore, they are easy to parameterize and can model a very wide family of probability distributions. Current EBMs approaches exploit the expressive power of neural networks to compute the energy function which have led to impressive results in a wide range of applications, including image generation, simultaneous

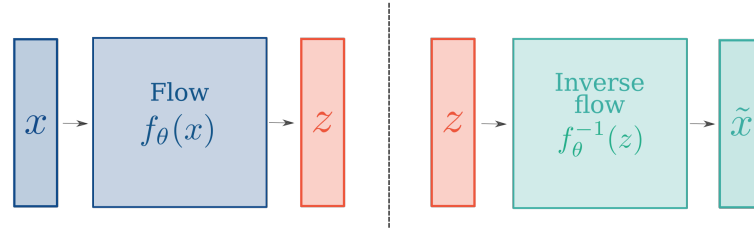


Figure 4.4 – Normalizing flows. A neural network f_θ –only composed by invertible layers– is trained to go from samples x in the input space, to latent variables z . The generative process simply consists in applying the inverse function f_θ^{-1} to latent variables z to obtain generated samples.

generation and classification, class-incremental classification, out-of-distribution detection, etc [200, 201].

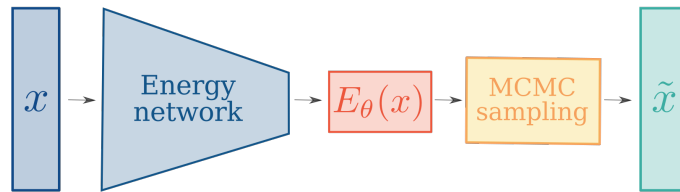


Figure 4.5 – Energy-based models (EBM). A neural network is used to estimate the energy function E_θ . Samples are generated by the means of MCMC sampling methods.

From the previous descriptions, we observe that these families of generative models differ on the way they estimate data distribution. Some of them estimate directly the likelihood function or a proxy of it, while others approximate the distribution in an implicit way. This has a direct impact on the trade-off to make between execution time, architecture to use and the objective function to optimize. Usually, learning the distribution implicitly comes with the advantage of getting more realistic and sharper generated images (like GANs), while the explicit expression of the likelihood function allows for other applications, like out-of-distribution detection (EBMs).

In what follows, we delve into Energy-based models. Indeed, their characteristics make them suitable for our EO applications: they are simple, since there is only one network to be trained (the energy estimator); they can generate sharp samples and, given infinite time, the procedure can generate true samples [200], they can learn distributions with multiple modes. Moreover, their simplicity –a simple neural network

that outputs a single scalar value that represents an energy function— allows to easily and naturally integrate label information into the generative model [201], by estimating the joint distribution $p(x, y)$ instead of $p(x)$; then there is no need for intricate modules for classification nor modifying the generative objective. This also leads to a natural extension of the model to semi-supervised learning, without introducing any changes to the network architecture. Moreover, EBMs have not been used for EO applications yet, so their potential in this field has still to be assessed.

4.2 Energy-based models

As introduced above, **energy-based models** capture dependencies between variables, $\mathbf{x} \in \mathcal{X}$, through a scalar function $E : \mathcal{X} \rightarrow \mathbb{R}$, known as the *energy function*. Learning an EBM consists in finding an energy function that associates low energy values to realistic configurations of x , and higher energy values to unrealistic ones. Then, the energy can be considered as a measure of compatibility of different configurations of variables.

EBMs can be interpreted as normalized probabilistic models using the Gibbs distribution, which expresses the density $p(\mathbf{x})$ as:

$$p(\mathbf{x}) = \frac{\exp(-E(\mathbf{x}))}{Z}, \quad (4.2)$$

where $Z = \int_{\mathcal{X}} e^{-E(\mathbf{x})}$ is a normalization constant.

Training EBMs comes with the advantage that the energy function parameterizes all the information about inputs. This alleviates the burden of computing or estimating the normalization constant Z , which is often intractable. Therefore, EBMs provide much more flexibility in the design—and thus the expressiveness—of learning models.

In this regard, EBMs have recently benefited from the expressive power of deep neural networks to model complex energy functions, with impressive results in generation, hybrid generation-classification and other applications [200, 201].

The standard way of training EBMs with deep learning today is by maximum likelihood estimation. Let p_{θ} be the probability density of an EBM, whose energy function, E_{θ} , is parameterized by a neural network of parameters θ . The density of the model, $p_{\theta}(\mathbf{x})$, can be fit to the distribution of data, $p_{\text{data}}(\mathbf{x})$, by maximizing the expected log-

likelihood function over all available samples.

$$\begin{aligned}\mathcal{L}_{\text{ML}}(\theta) &:= \mathbb{E}_{\mathbf{x} \sim p_{\text{data}}}[\log p_{\theta}(\mathbf{x})] \\ &= \mathbb{E}_{\mathbf{x} \sim p_{\text{data}}}[-E_{\theta}(\mathbf{x})] - \log Z_{\theta}.\end{aligned}\quad (4.3)$$

A common way of optimizing this objective is by gradient descent. Thus, we need to compute the gradient of the log-likelihood in Eq. (4.3).

$$\nabla_{\theta} \mathcal{L}_{\text{ML}}(\theta) = \nabla_{\theta} \mathbb{E}_{\mathbf{x} \sim p_{\text{data}}}[-E_{\theta}(\mathbf{x})] - \nabla_{\theta} \log Z_{\theta}.\quad (4.4)$$

The gradient of $\log Z_{\theta}$ is computed as:

$$\begin{aligned}\nabla_{\theta} \log Z_{\theta} &= \frac{1}{Z_{\theta}} \nabla_{\theta} \int_{\mathcal{X}} \exp(-E_{\theta}(\mathbf{x})) \\ &= - \int_{\mathcal{X}} \frac{\exp(-E_{\theta}(\mathbf{x}))}{Z_{\theta}} \nabla_{\theta} E_{\theta}(\mathbf{x}) \\ &= - \int_{\mathcal{X}} \nabla_{\theta} [E_{\theta}(\mathbf{x})] p_{\theta}(\mathbf{x}) \\ &= - \mathbb{E}_{p_{\theta}(\tilde{\mathbf{x}})}[\nabla_{\theta} E_{\theta}(\tilde{\mathbf{x}})].\end{aligned}\quad (4.5)$$

And thus, from Eq. (4.4) and Eq. (4.5), the gradient of the log-likelihood can finally be expressed as:

$$\nabla_{\theta} \mathcal{L}_{\text{ML}}(\theta) = \mathbb{E}_{p_{\theta}(\tilde{\mathbf{x}})}[\nabla_{\theta} E_{\theta}(\tilde{\mathbf{x}})] - \mathbb{E}_{p_{\text{data}}(\mathbf{x})}[\nabla_{\theta} E_{\theta}(\mathbf{x})].\quad (4.6)$$

And the maximum likelihood objective in Eq. (4.3) can be expressed as¹:

$$\mathcal{L}_{\text{ML}}(\theta) = \mathbb{E}_{p_{\theta}(\tilde{\mathbf{x}})}[E_{\theta}(\tilde{\mathbf{x}})] - \mathbb{E}_{p_{\text{data}}(\mathbf{x})}[E_{\theta}(\mathbf{x})].\quad (4.7)$$

To compute the gradient expressed in Eq. (4.6), one needs to be able to sample from the model distribution p_{θ} , which is not possible because of its complexity. Current approaches approximate p_{θ} using MCMC methods, like Gibbs sampling [202], Hamiltonian Monte Carlo [203] or Langevin dynamics [204]. This allows us to approximately optimize the log-likelihood objective and generate samples from the model. In particular, Langevin dynamics –and more precisely, Stochastic gradient Langevin dynamics

1. In practice, the log-likelihood loss is computed by this expression

(SGLD)– has been widely used for training EBMs [200] as it uses gradient information and initializes Markov chains from random noise for improved mixing, which allows to perform more efficient sampling.

The main idea behind SGLD is to generate low-energy data points according to the current model. It is very similar to stochastic gradient descent, since we start with randomly sampled points, then we find the direction of minimum energy (by the means of the gradient) and take a step towards that direction. By repeating this process, one eventually reaches points of minimum energy. Noise is injected into this procedure to ensure that we sample points that are around the modes of the distribution.

4.2.1 Joint energy-based models (JEM)

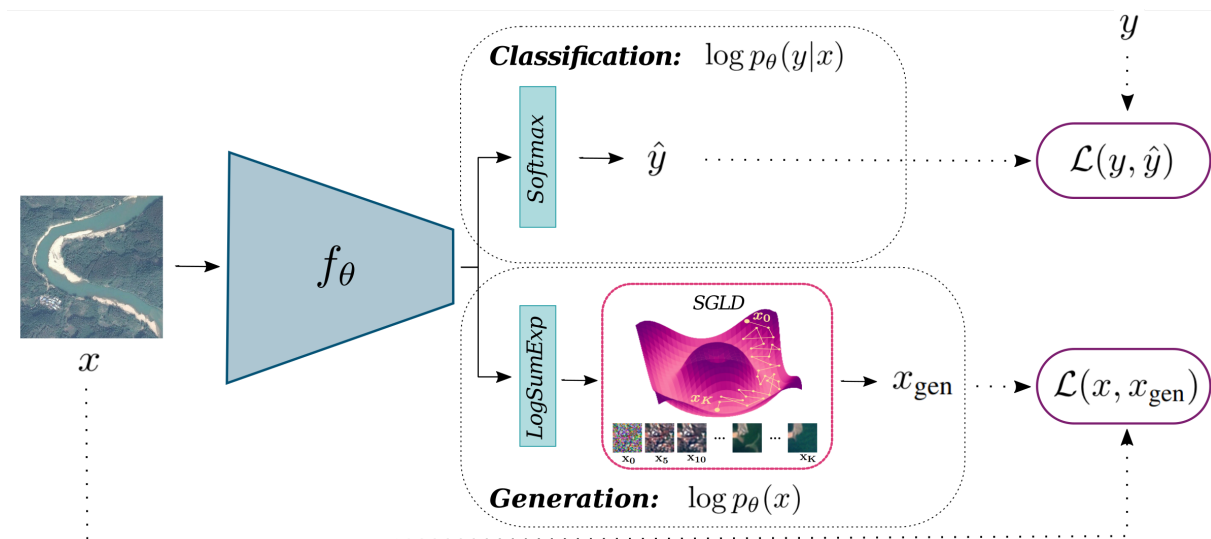


Figure 4.6 – JEM overview. In a nutshell, an input image x passes through a neural network f_θ . Then, the pipeline splits into two modules: (i) a classification module that applies a softmax function to $f_\theta(x)$ to obtain class scores and computes the classification loss (cross-entropy), and (ii) a generation module that computes the energy E_θ from Eq. (4.10) (LogSumExp), then runs a finite Stochastic Gradient Langevin Dynamics (SGLD) chain (Eq. (4.12)), drawing samples from $p_\theta(x)$ and uses them to compute the log-likelihood loss. The sum of both loss terms (Eq. (4.11)) is then optimized by backpropagation.

Joint energy-based models (JEM) [201] have been recently presented to extend a standard classification neural network into an hybrid discriminative-generative model, by simply re-interpreting the outputs of the classifier.

Let $f_\theta : \mathbb{R}^D \rightarrow \mathbb{R}^K$ be a classification neural network, parameterized by θ , with K the number of classes and D the input's dimension. The fundamental idea of JEM is to express the joint distribution of images (\mathbf{x}) and labels (y) as a joint energy-based model:

$$p_\theta(\mathbf{x}, y) = \frac{\exp(f_\theta(\mathbf{x})[y])}{Z_\theta}, \quad (4.8)$$

where the joint-energy function is parameterized by the neural network: $E_\theta(\mathbf{x}, y) = -f_\theta(\mathbf{x})[y]$. $f_\theta(\mathbf{x})[y]$ is the y -th entry of $f_\theta(\mathbf{x})$ and Z_θ the normalizing constant of the model.

By marginalizing Eq. (4.8) above, we obtain the distribution $p_\theta(\mathbf{x})$ expressed as:

$$p_\theta(\mathbf{x}) = \sum_{y=1}^K p_\theta(\mathbf{x}, y) = \frac{\sum_{y=1}^K \exp(f_\theta(\mathbf{x})[y])}{Z_\theta}. \quad (4.9)$$

From Eq. (4.9), one may observe that the distribution $p_\theta(\mathbf{x})$ is also an energy-based model, with the energy given by:

$$E_\theta(\mathbf{x}) = -\log \left(\sum_{y=1}^K \exp(f_\theta(\mathbf{x})[y]) \right). \quad (4.10)$$

The JEM model is then trained to maximize the joint log-likelihood, $\log p_\theta(\mathbf{x}, y)$, which can be factorized as:

$$\log p_\theta(\mathbf{x}, y) = \log p_\theta(\mathbf{x}) + \log p_\theta(y|\mathbf{x}). \quad (4.11)$$

As shown below, Eq. (4.11) is the key to obtain a joint discriminative-generative model.

Generation The first term in Eq. (4.11), $\log p_\theta(\mathbf{x})$, corresponds to the generative part of the model. It is trained as an energy-based model by approximating the gradient $\nabla_\theta \mathcal{L}_{\text{ML}}(\theta)$ (Eq. (4.6)) using a sampler based on Stochastic Gradient Langevin Dynamics (SGLD) [200] and thus, generates samples following:

$$\mathbf{x}_{i+1} = \mathbf{x}_i - \frac{\alpha}{2} \nabla_{\mathbf{x}} E_\theta(\mathbf{x}_i) + \varepsilon, \quad \mathbf{x}_0 \sim p_0(\mathbf{x}), \quad (4.12)$$

with $\varepsilon \sim \mathcal{N}(0, \alpha)$ and $p_0(\mathbf{x})$ usually a uniform distribution, and α a step-size following a polynomial decaying.

This allows to generate samples as an iterative process, starting from noise, following the energy gradient's direction, as shown in Fig. 4.7.

In practice, the loss to optimize is the maximum log-likelihood loss as expressed in Eq. (4.7), with the energy function E_θ in Eq. (4.10).



Figure 4.7 – Generative process by SGLD, following Equation (4.12).

Classification The second term is related to $p_\theta(y|\mathbf{x})$, which matches the softmax output of a usual classifier when it is written as $p_\theta(y|\mathbf{x}) = p_\theta(\mathbf{x}, y) / p_\theta(\mathbf{x})$. Thus it can be simply optimized using the cross-entropy loss, as a standard classification neural network.

Figure 4.6 illustrates how JEM works in practice. An input image x passes through a neural network f_θ , which outputs $f_\theta(\mathbf{x}) \in \mathbb{R}^K$. Then, the pipeline splits into two modules: (i) a classification module (Fig. 4.6 top) that applies a softmax function to $f_\theta(\mathbf{x})$ to obtain class scores and computes the classification loss (cross-entropy), and (ii) a generation module (Fig. 4.6 bottom) that computes the energy E_θ from Eq. (4.10) (Log-SumExp), then runs a finite SGLD chain (Eq. (4.12)), drawing samples from $p_\theta(\mathbf{x})$ and uses them to compute the log-likelihood loss. The sum of both loss terms (Eq. (4.11)) is then optimized by backpropagation.

4.2.2 Semi-supervised learning with JEM

Moreover, JEM, as described above, also allows to extend a conventional classifier to semi-supervised learning in a very natural way [205].

Indeed, if labels are available, one can optimize the main objective $\log p_\theta(\mathbf{x}, y)$ as in Eq. (4.11), otherwise one may simply marginalize it out and optimize $\log p_\theta(\mathbf{x})$ only. In practice, following the scheme in Fig. 4.6, this means that for labeled samples the network is updated as described above (Section 4.2.1), but unlabeled samples only go through the generation module (bottom section of Fig. 4.6) to update the network.

We have recalled here the main concepts of JEM, a recent energy-based model for joint generation and classification of images. However, to the best of our knowledge, the relevance of such energy-based models to deal with EO data has not been studied

yet. We report in the next section some experiments we conducted with JEM to address various applications of high interest in remote sensing.

4.3 Experiments

Since JEM is a multifaceted model, in this section we explore its capacities in various tasks, including: classification, generation, semi-supervised classification, out-of-distribution detection and land cover mapping. In Table 4.1, we compare JEM to other models that perform well on each task, however none of them is as versatile as JEM, being limited to one or two tasks to perform simultaneously.

We perform experiments by training our models on several publicly available EO datasets for scene classification²: EuroSAT [124], So2Sat LCZ42 [125], Aerial Image Dataset [126] and UCMerced [127].

The **EuroSAT Dataset** comprises patches from Sentinel-2 images over 34 countries in Europe. Each patch is labeled with one of 10 land cover/land use classes (e.g. industrial, residential, highway, pasture, forest, etc.). Classes are well-balanced, with 2,000 to 3,000 examples per class, 80% of which are kept for training. We use the EuroSAT RGB version.

The **So2Sat LCZ42 Dataset** is composed of Sentinel-1 and Sentinel-2 image patches over 42 locations over the globe. Patches are labeled according to the Local Climate Zones scheme (LCZ), with 17 categories. It is worth to mention that the training set and testing set are geographically independent, containing images from different locations. This makes this dataset particularly difficult, because models need to be sufficiently robust to generalize well on the test data. For our experiments, we only make use of the RGB Sentinel-2 bands (B04, B03, B02), as in EuroSAT.

The **Aerial Image Dataset (AID)** consists of 10,000 optical aerial images from different countries around the world, labeled within 30 scene classes. Original RGB tiles are of size 600px × 600px. Due to the computing time of JEM, we have resized them to 64px × 64px during training.

The **UCMerced Dataset** is a small-size dataset and has been widely used for the evaluation of aerial scene classification. It contains 2,100 aerial ortho-images from different regions of USA. Each image is labeled with one of the 21 land use classes. Original

2. While the two first have already been used in previous chapters, they are introduced again here for the reader only interested in generative models.

Model	Classification	Generation	Semi-supervision	OOD detection
Wide-ResNet	✓	✗	✗	✗
VAE	✗	✓	✗	✗
GAN	✗	✓	✗	✗
BerundaNet	✓	✗	✓	✗
FixMatch	✓	✗	✓	✗
JEM	✓	✓	✓	✓

Table 4.1 – Models comparison. JEM is the only model able to perform all these tasks simultaneously.

Type	Model	Classification Accuracy (↑)	Generation FID (↓)	Generation KID (↓)
Discriminative	Wide-ResNet	97.56 ±0.52 %	✗	✗
Hybrid	JEM	97.42±0.19 %	122.1	0.06
Generative	VAE	✗	215.4	0.14

Table 4.2 – Classification and generation scores of models trained on EuroSAT. Comparison of JEM with respect to a purely supervised model (Wide-ResNet-28-10) and a purely generative model (VAE). Note that JEM is the only model that can provide both classification and generation scores. Best scores in bold.

256px × 256px tiles have been resized to 64px × 64px for JEM training.

For evaluating out-of-distribution detection and other tasks of interest, we use in addition several public EO datasets: ISPRS Potsdam [133], OSCD dataset [206], DFC 2017 [150] and BigEarthNet [91].

Implementation details. Following [201], we perform our experiments using a Wide-ResNet-28-10 architecture [207], with no batch normalization. We train our networks with the Adam optimizer [208], during 200 epochs, following the JEM training scheme.

Moreover, we adopt a hold-out evaluation method, defining a training and a test set (80% and 20% of data, respectively, for all datasets, except So2Sat LCZ42 where train and test partitions are already defined). Additionally, 10% of the training set was used as validation partition during training. This is especially important when training on very few labeled data to adopt an early stopping strategy and avoid overfitting.

Pytorch [209] is used for all implementations.

Labeled samples/class	% of labels	Wide-ResNet	BerundaNet	FixMatch	JEM
2000 on avg.	100%	97.56 \pm 0.52	96.90 \pm 0.67	98.81 \pm 0.06	97.42 \pm 0.19
100	\sim 5%	86.36 \pm 0.26	74.78 \pm 2.01	97.83 \pm 0.12	86.23 \pm 0.80
20	\sim 1%	62.93 \pm 1.01	54.25 \pm 2.41	95.78 \pm 0.99	<u>69.11</u> \pm 1.18
10	\sim 0.5%	52.33 \pm 1.59	46.84 \pm 1.83	94.95 \pm 1.12	<u>61.60</u> \pm 1.49
5	\sim 0.25%	43.83 \pm 3.18	39.80 \pm 1.51	94.45 \pm 1.29	<u>54.79</u> \pm 3.55
1	\sim 0.05%	28.02 \pm 0.97	32.77 \pm 1.05	67.46 \pm 4.67	<u>36.86</u> \pm 1.11

Table 4.3 – Classification results on EuroSAT (Accuracy [%] \uparrow). Comparison with a purely supervised method (Wide-ResNet), a multi-task semi-supervised network (BerundaNet) and a semi-supervised method based on consistency regularization (FixMatch), trained on the same number of labeled samples. Grey cells indicate model leveraging unlabeled data. Best scores in bold, second best underlined (when significant).

4.3.1 Joint classification and generation with JEM

In this section we show that this new training paradigm allows to get an hybrid model, with competitive performances in both tasks, classification and generation.

Wide-ResNet is trained as a usual classifier (with cross-entropy loss), while JEM is trained as described in Section 4.2.1. We compare the generative performance with a standard VAE [210]. Results on the EuroSAT dataset are summarized in Table 4.2.

Given uncertainty measured by standard deviation, JEM results reach the same level of performances as classification-only Wide-ResNet and previously reported classification results on EuroSAT, namely ResNet-50 and GoogLeNet with 98.6% and 98.2% of overall accuracy respectively [124]. The small difference observed might be explained by the intrinsic regularization of the multi-task JEM model. Furthermore, [124] does not specify a training and test partition, which might also explain the discrepancy with our results. In terms of generation, we rely on the Fréchet Inception Distance (FID) [211] and the Kernel Inception Distance (KID) [212] to evaluate the quality of generated samples. According to these metrics, JEM generated samples are superior to VAE samples.

Fig. 4.8 shows some class-conditional examples generated by the network after being trained on the EuroSAT dataset, with different settings. Each row represents a class in the dataset. First two columns show real samples from the dataset, third and fourth

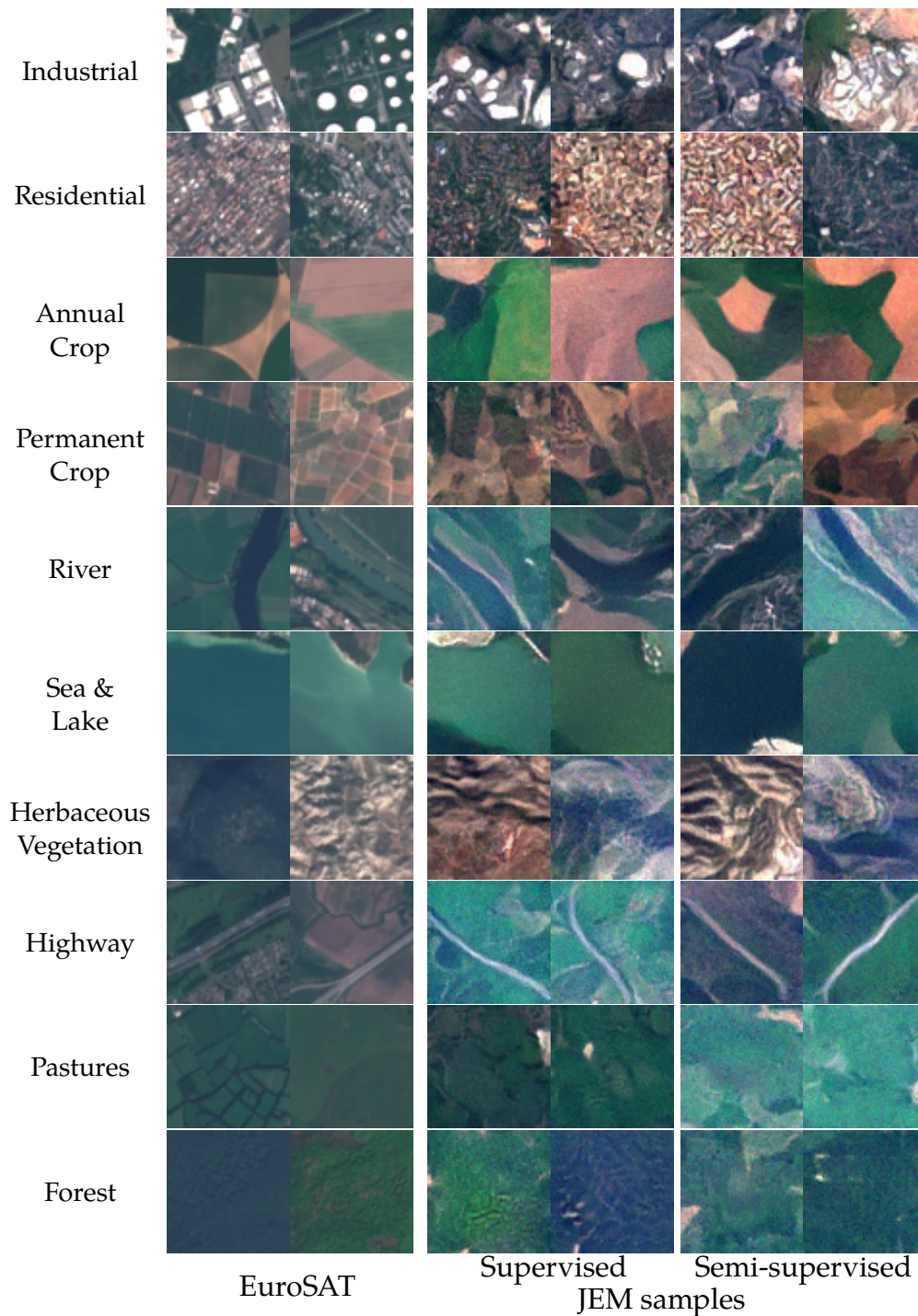


Figure 4.8 – Class-conditional samples generated by Joint Energy-based Model (JEM) trained on the EuroSAT dataset. First two columns contain real EuroSAT samples. Third and fourth columns present JEM-generated samples trained on all training samples. Last two columns show samples generated following a semi-supervised learning strategy, with 100 labeled samples per class.

columns present JEM-generated samples trained on the whole EuroSAT dataset and last two columns display JEM-generated samples with the model trained in a semi-supervised manner with 100 labeled samples per class (and the rest of the dataset as unlabeled data, more details in Section 4.3.2). From these examples, we observe that JEM captures the data distribution properly, since generated samples are extremely similar to real EuroSAT samples regardless of the fraction of annotated examples available for training. Moreover, the model is capable to produce samples for every class in the dataset, with a large variety of images per class.

However, some classes remain difficult to apprehend, e.g. forests or sea and lakes. This might be due to the lack of texture on these images. Industrial buildings (first row in Fig. 4.8) would require finer and more rectangular outlines to correctly match industrial buildings in the EuroSAT dataset. On the other hand, JEM is able to handle impressively images of highways, rivers and different types of fields. Indeed, generated samples of these classes are remarkably similar to real images. This shows that the model is able to learn the true distribution behind the dataset and leads to compelling applications. Synthetic examples generated from the learned data distribution may be used for simulation or even for training new models.

4.3.2 Semi-supervised classification with JEM

In this section we perform semi-supervised classification and show the potential of JEM in extreme settings when very few labeled samples are available.

We train the JEM model with a small subset of labeled examples and the entire dataset as unlabeled data, following the approach described in Section 4.2.2. We vary the number of labeled samples per class on which the model is trained, and compare our results with three baselines: the fully supervised Wide-ResNet, BerundaNet [J2] (semi-supervised), and FixMatch [73] (semi-supervised). Wide-ResNet, as a supervised method, is trained on labeled data only, while BerundaNet, FixMatch and JEM are trained similarly on the whole dataset, using labels when available. Table 4.3 summarizes our results on the EuroSAT dataset.

First row in Table 4.3 presents completely supervised results on the entire training set, as an upper bound for the semi-supervised strategies. We observe that all methods are on par in terms of performance, FixMatch being slightly better. We observe from the following rows that FixMatch, being especially designed to tackle the semi-supervised classification problem, is superior to all methods and performs remarkably well, even in

Dataset	Labeled samples/class	% of labels	Wide-ResNet	JEM
So2Sat	~ 20000	100%	50.93 ± 0.16	54.60 ± 0.35
	1000	~ 5%	44.17 ± 0.40	48.59 ± 0.58
	200	~ 1%	35.45 ± 0.17	42.43 ± 0.47
	100	~ 0.5%	30.90 ± 0.35	38.71 ± 0.64
AID	~ 300	100%	78.71 ± 0.08	74.11 ± 0.24
	20	~ 7%	41.07 ± 1.87	50.23 ± 0.69
	13	~ 5%	34.46 ± 0.59	44.49 ± 0.65
	3	~ 1%	17.38 ± 0.32	25.68 ± 0.65
	1	~ 0.5%	9.98 ± 0.36	16.21 ± 0.58
UCMerced	80	100%	81.71 ± 0.72	80.49 ± 1.67
	10	~ 12.5%	45.41 ± 0.43	48.91 ± 0.42
	4	~ 5%	26.99 ± 1.24	34.16 ± 1.78
	1	~ 1%	14.34 ± 1.88	24.31 ± 1.87

Table 4.4 – Classification results on different EO datasets (Accuracy [%] ↑). Grey cells indicate model leveraging unlabeled data. Best scores in bold (when significant).

extreme situations when very few labeled data are available. In the case of BerundaNet and JEM, there is a point where they perform considerably better than Wide-ResNet. In the case of JEM, there is no significant difference with respect to the Wide-ResNet performance in the 5% and 100% labeled samples per class regime, however the advantage of JEM becomes tangible as soon as the model is trained with 1% of labeled samples or less, with a performance gap varying from 6.2% to 10.7% of accuracy. Moreover, in the semi-supervised setting, JEM is always superior to BerundaNet. This difference might be explained by the way these methods leverage unlabeled samples during training. Indeed, BerundaNet uses them to compute a regularization term through a secondary task (here reconstruction), while JEM uses unlabeled samples to estimate their underlying distribution, which might contain valuable information for classification.

These results show that: first, the energy function can be learned from unlabeled data as well as labeled data; and second, if the image distribution $p_{\theta}(\mathbf{x})$ is well estimated, it is easier then to estimate the conditional distribution $p_{\theta}(y|\mathbf{x})$ from a small set of annotated training samples.

Furthermore, even if FixMatch has undeniably better performances in the semi-

supervised settings, it is worth to notice that it was especially designed to perform semi-supervised classification, and cannot perform other tasks like OOD detection nor generation (see Table 4.1). On the contrary, JEM is a versatile model that can perform several tasks simultaneously. Moreover, it could be optimized to achieve better results on semi-supervised learning, for instance, by integrating FixMatch features such as massive data augmentation strategies and consistency regularization³.

Additionally, we compare JEM and Wide-ResNet on three well-known benchmarks for scene classification, So2Sat LCZ42, AID and UCMerced. Results are summarized on Table 4.4.

On AID and UCMerced, results confirm what we observed previously on EuroSAT: in the supervised setting, when all labels are available for the training data, Wide-ResNet is slightly superior to JEM, because of the intrinsic regularization of the latter. However, when few labels are available, JEM has considerably better classification performance. On the other hand, results over So2Sat –a more realistic and large-scale dataset– not only confirm the tendency observed on EuroSAT, but the superiority of the JEM model over Wide-ResNet is even more consistent, including the supervised setting. This is explained by the existing domain gap between training data and testing data in So2Sat, due to different geographic locations. Indeed –as we observed in our experiments in Chapter 2– standard discriminative classifiers, like Wide-ResNet, are prone to lack robustness to distribution shifts. However, learning the underlying distribution of the data by a generative model such as JEM helps to overcome this issue and sets a starting point to bridge the performance gap when dealing with domain shifts.

Model calibration

Beyond classification scores, an important and desirable feature of models is the calibration. A model is said to be calibrated if its output confidence, usually measured as $\max_y p(y|\mathbf{x})$, coincides with its expected accuracy⁴. Therefore, a calibrated model is more informative, being able to provide the uncertainty associated to a prediction.

We thus evaluate and compare the calibration of the supervised (Wide-ResNet) and semi-supervised models (FixMatch and JEM). In particular, we study the 100-labeled-samples-per-class and the 10-labeled-samples-per-class settings. Figure 4.9 shows the calibration curves for both experiments. A perfectly calibrated classifier should match

3. Which we will explore later, in Section 4.3.5.

4. In practice, the calibration is estimated on an evaluation set. Confidence of the model is computed for each sample on this set, then samples are binned according to their confidence score. Finally, the mean accuracy is computed for each bin, which allows us to plot a calibration curve.

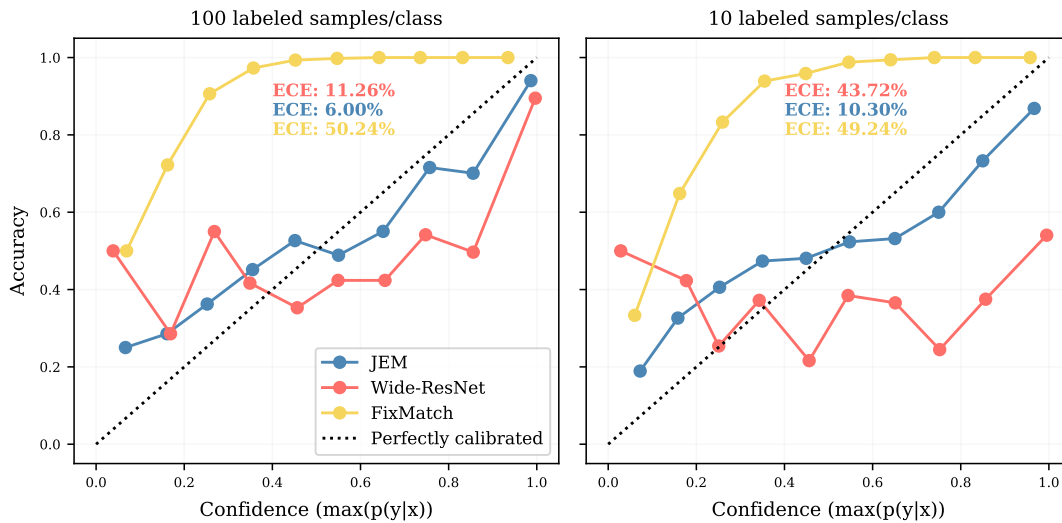


Figure 4.9 – Calibration curves for supervised Wide-ResNet and semi-supervised JEM and FixMatch trained on EuroSAT dataset. Left: trained on 100 labeled samples per class. Right: trained on 10 labeled samples per class. ECE: Expected Calibration Error (\downarrow).

the straight line $y = x$. We can observe that, in both settings, JEM is the model with best calibration, FixMatch being very underconfident and Wide-ResNet being overconfident.

We quantitatively verify this by computing a usual metric for calibration: the Expected Calibration Error [213] (ECE) score, for both settings. The obtained ECE scores are 11.26%, 50.24% and 6.00% for Wide-ResNet, FixMatch and JEM, respectively, in the case of 100-labeled-samples-per-class; and 43.73%, 49.24% and 10.22% in the extreme setting of 10-labeled-samples-per-class. Since a perfect ECE is equal to zero, these scores confirm that the semi-supervised JEM model is better calibrated, the difference being flagrant in extreme conditions (very few labels). FixMatch exhibits very poor calibration properties. Therefore, unlabeled data regularization, by learning the data distribution, comes with the advantage of allowing for more informative predictions.

4.3.3 Out-of-distribution analysis

Out-of-distribution detection (OOD) [214] refers to the task of recognizing significantly different or anomalous examples, with respect to the ones seen during training. Asserting the capacity of a model to correctly classify a sample from a new domain is

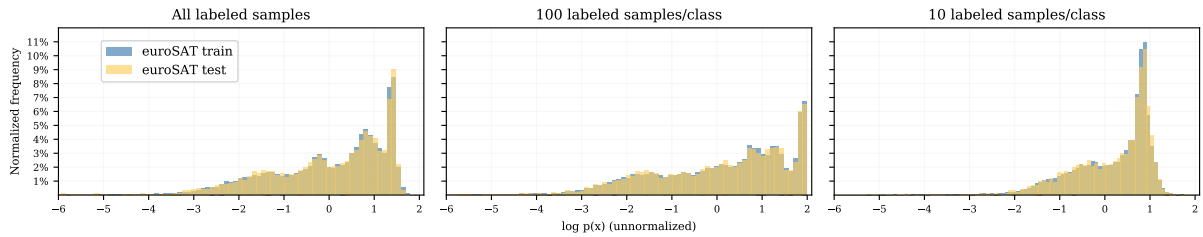


Figure 4.10 – JEM log-likelihood (unnormalized) histograms for EuroSAT dataset. Stability of the estimated energy function. Supervised vs. Semi-supervised with 100 labeled samples and 10 labeled samples per class comparison. We observe that the values of the unnormalized log-likelihood are comparable, regardless the amount of labeled data available during training.

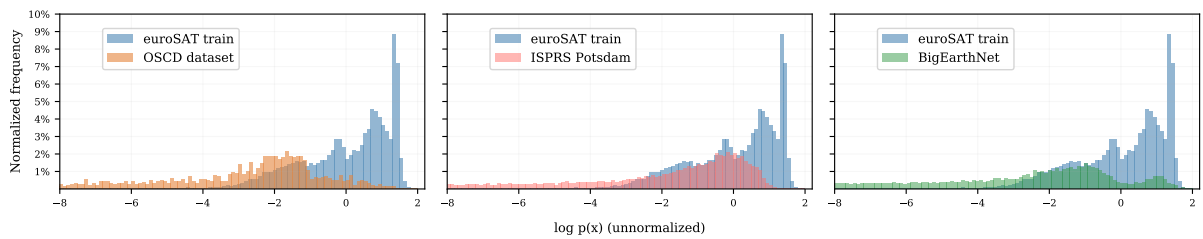


Figure 4.11 – Out-of-Distribution detection on different public EO datasets. Unnormalized log-likelihood values computed through the supervised model.

a very important and desirable feature, especially in applications which involve real-world decisions.

In this section, we assess the capacity of our model to assess global out-of-distribution analysis, i.e. if an entire dataset can be considered *in-distribution* with respect to the learned distribution. In this regard, we compare the histograms of the unnormalized log-likelihood (i.e. $-E(x)$) values of the EuroSAT training set with the obtained histograms for different public datasets.

Supervised vs. semi-supervised energy function

Figure 4.10 presents the unnormalized log-likelihood histograms for the EuroSAT dataset in the fully-supervised JEM setting (left) and two semi-supervised settings: training with 100 labeled samples per class (center) and with 10 labeled samples per class (right). In all cases, the histogram profiles of the training and test partition match perfectly, which means that, as expected, there is no shift of the estimated distribution from EuroSAT train to EuroSAT test.

Moreover, we observe that the log-likelihood distribution estimated by the models is very similar, showing that the energy is not linked to the labels, but to the data.

Comparing datasets

On the other hand, Figure 4.11 shows the unnormalized log-likelihood histograms of 3 public EO datasets: OSCD dataset [206], ISPRS Potsdam [133] and BigEarthNet [91], obtained after training the model on the whole EuroSAT [124] training set.

We observe that for these datasets, the histogram profile does not exactly match the one of the EuroSAT training data. Actually, values of the unnormalized $\log p(\mathbf{x})$ can be extremely small, which can be interpreted as the samples from these datasets are not likely to come from the distribution learnt from EuroSAT. We can confirm this observation by computing the Kullback-Leibler (KL) divergence with respect to the distribution of the EuroSAT train histogram. Indeed, while KL value for EuroSAT test data is 0.27; the other datasets KL values are 28.2, 25.6 and 26.3 for Potsdam, OSCD and BigEarthNet, respectively. In view of this, more information would be needed for the model to correctly represent those datasets that differ on location, resolution or appearance.

Finally, it is interesting to notice that the distribution that differs the most is ISPRS Potsdam, the only dataset with a different resolution. This might imply that resolution is an important factor for domain adaptation.

4.3.4 Application to land cover mapping

Land cover mapping is an interesting application of JEM on new unseen domains as detailed in the following sections.

Patch-wise classification

We apply our EuroSAT-trained models –including Wide-ResNet, supervised and semi-supervised JEM– to unseen OSCD tiles. To do so, the tiles are split into 64×64 patches which go through the already trained network to obtain the corresponding class per patch, leading to a patch-wise classification map.

We observe in Figure 4.12 the results on two locations from OSCD: Beirut and Rio de Janeiro. The maps produced by the classifier are, in general, globally correct and

retrieve various densities of urban and green areas. As expected, the quality of predictions deteriorates as the number of labeled samples decreases. Indeed, supervised Wide-ResNet and JEM predictions are both plausible land cover maps for these locations. The map of JEM semisup-100 is still trustworthy, while 10 labeled samples per class seem not enough to train an accurate model.

Similarly, we apply the So2Sat LZC42-trained models to the unseen tile of Rome from DFC2017. Since So2Sat LZC42 is composed of 32×32 images, the Rome tile is also split in 32×32 patches to pass through the network. Figure 4.13 presents our patch-wise classification maps. As before, the maps are reasonable, JEM being more accurate than Wide-ResNet to recognize low plants, where Wide-ResNet overestimates heavy industry.

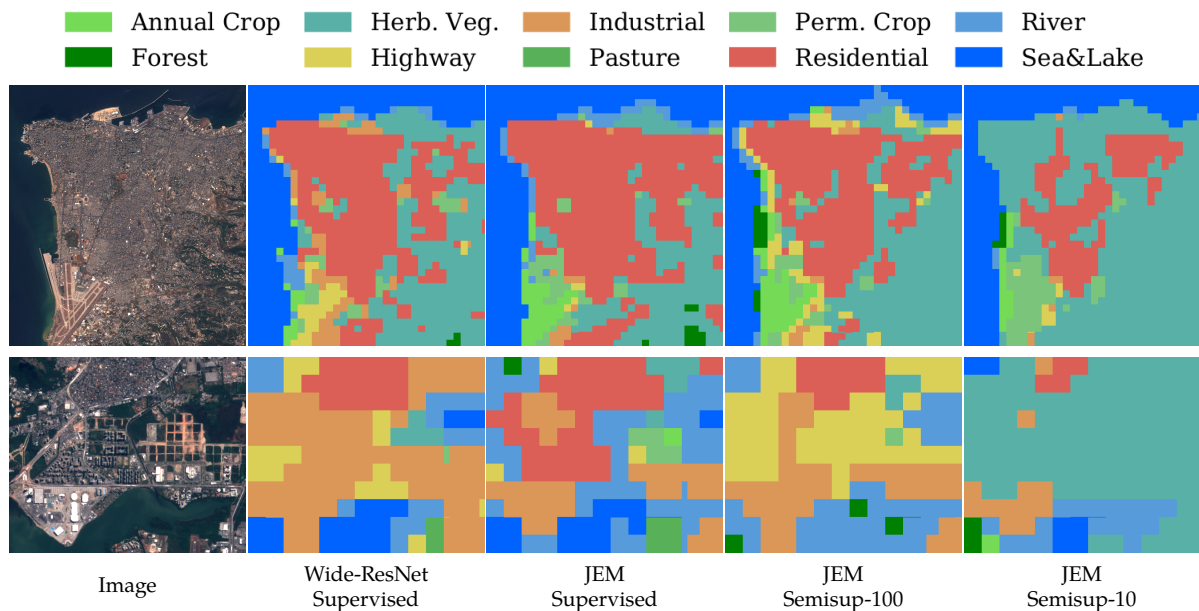


Figure 4.12 – Semantic maps on never-seen OSCD cities. Top: Beirut. Bottom: Rio de Janeiro. Supervised indicates models trained on the entire EuroSAT dataset. Semisup- x is JEM trained with a semi-supervised strategy with x labeled samples per class. No semantic segmentation ground truth is provided with this dataset.

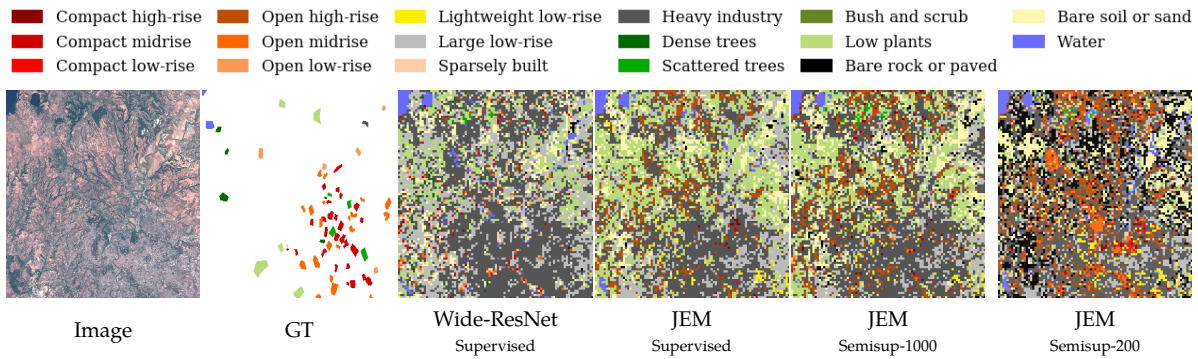


Figure 4.13 – Semantic maps on never-seen DFC 2017 tile of Rome. Supervised indicates models trained on the entire So2Sat dataset. Semisup- x is JEM trained with a semi-supervised strategy with x labeled samples per class.

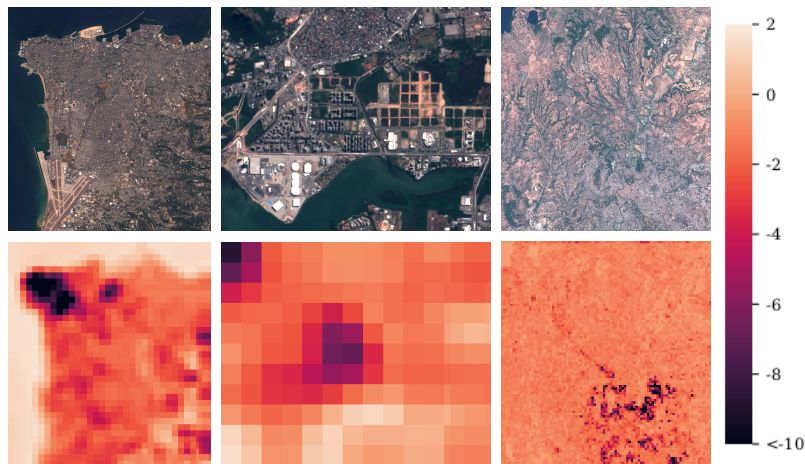


Figure 4.14 – Confidence maps obtained by JEM on never-seen OSCD and DFC2017 tiles. Confidence is measured as the unnormalized $\log p(\mathbf{x})$, From left to right: Beirut, Rio de Janeiro and Rome.

Confidence maps

The major advantage of JEM over a standard classifier such as Wide-ResNet is its capacity to estimate the underlying data distribution through the energy function. We can use the unnormalized log-likelihood value as a proxy for the confidence of the model's prediction. Indeed, if the model assigns a high value of log-likelihood to an image it could be considered as *in-distribution*, and thus the model's prediction should be pertinent. Conversely, if the model's log-likelihood on a sample is low, we could consider it as *out-of-distribution* and be more cautious with respect to its prediction.

Figure 4.14 shows the confidence maps obtained by the supervised JEM over the OSCD tiles (trained on EuroSAT) and over Rome tile from DFC2017 (trained on So2Sat). We observe that the confidence of the model varies across the patches. Indeed, on OSCD, the model is more confident on scenes representing water or fields, while it is considerably less confident in residential and industrial areas, which are more likely to be different from training European cities from the EuroSAT dataset. In the case of Rome, the model is less confident in general, and in particular on the compact zones (according to the ground-truth in Fig. 4.13).

4.3.5 Can we combine FixMatch and JEM?

As mentioned in Section 4.3.2, JEM can still be optimized to achieve better results on semi-supervised learning. One possible way to achieve this is by integrating FixMatch features into the model, such as data augmentation strategies and consistency regularization.

A very simple and yet effective procedure to combine FixMatch and JEM is described in the following:

- (i) Train a semi-supervised FixMatch model with very few labeled data;
- (ii) Generate (pseudo-) labels for the rest of the training set with the model trained on the previous step. Since FixMatch is very efficient, these pseudo-labels should be quite accurate.
- (iii) Finally, train JEM on this pseudo-labeled training set and apply it to the test set.

The results of this first experiment are shown in Table 4.5. Indeed, we observe that the FixMatch&JEM procedure described above outperforms JEM alone. Therefore, we have obtained a better semi-supervised classifier, and –at the same time– kept a robust

Labeled samples/class	% of labels	JEM	FixMatch & JEM
100	~ 5%	86.23 \pm 0.80	85.94 \pm 0.48
20	~ 1%	69.11 \pm 1.18	84.96 \pm 0.57
10	~ 0.5%	61.60 \pm 1.49	84.89 \pm 0.76
5	~ 0.25%	54.79 \pm 3.55	82.46 \pm 0.98
1	~ 0.05%	36.86 \pm 1.11	56.78 \pm 1.31

Table 4.5 – Classification results of a first combination of FixMatch & JEM on EuroSAT (Accuracy [%] \uparrow). Comparison with JEM. Grey cells indicate model leveraging unlabeled data. Best scores in bold (when significant).

generative model, able to synthesize new data, perform OOD detection, and other applications as we have shown all along this section (Section 4.3). While FixMatch alone reaches higher classification accuracy (see Table 4.3), it does not offer these appealing properties.

Future works on combining FixMatch and JEM should considerate training a model end-to-end. For instance, one could design a combined pipeline, where JEM is integrated into the FixMatch algorithm by using the weakly augmented inputs of FixMatch to optimize the generative objective of JEM (in addition to the corresponding FixMatch loss terms). Thus, the objective function to optimize would be composed of three terms: $\mathcal{L} = \mathcal{L}_{\text{sup}} + \mathcal{L}_{\text{consistency}} + \mathcal{L}_{\text{ML}}$, where \mathcal{L}_{sup} is the supervised classification loss, $\mathcal{L}_{\text{consistency}}$ is the consistency regularization from FixMatch, defined by Eq. (3.25), and \mathcal{L}_{ML} is the generative loss proper to JEM, from Eq. (4.7). One of the main challenges of this approach would be the joint optimization: finding the right hyper-parameters to achieve the best of both worlds.

4.4 Limitations

Training Energy-based Models by maximum likelihood can be very challenging. Indeed, the gradient estimators used to estimate log-likelihood are considerably unstable and prone to diverging during training, this is why hyperparameters must be chosen carefully. Moreover, MCMC-like iterative sampling increases training time linearly with the image size. This may be prohibitive when dealing with large images, which is likely the case in remote sensing applications. This is why we decided to resize AID

and UCMerced images for our experiments in Section 4.3.2.

Despite these limitations, we strongly believe that the remote sensing community might deeply benefit from the multiple applications of EBMs, that we tried to bring forward in this work. We believe that there is still much progress to make to improve and optimize EBMs' training, just as the community has achieved great progress on GANs' training in only a few years.

4.5 Perspectives: semantic segmentation with JEM

As we have shown, JEM have several high-potential applications in remote sensing. Therefore –and even if we have shown that it can be applied to large tiles by patch-wise classification– it would be useful to extend this model to semantic segmentation. In what follows, we present our preliminary work on this extension, mostly theoretical, practical experiments being left for future work.

The most explicit difference between semantic segmentation and scene classification is the output's dimension. While classification algorithms take as input an image and output a label $y \in \{1, \dots, K\}$ (with K the number of possible classes), a semantic segmentation algorithm takes as input an image $X \in \mathbb{R}^{W \times H \times 3}$ and outputs a segmentation map $Y \in \{1, \dots, K\}^{W \times H}$, in other words, a class label for each pixel in the input image.

Let $f_\theta : \mathbb{R}^{W \times H \times 3} \rightarrow \mathbb{R}^{W \times H \times K}$ be the function learned by a segmentation network, parameterized by θ . Let $f_\theta(X)^i$ be the value of the network's output at pixel i , then $f_\theta(X)^i \in \mathbb{R}^K$ and $f_\theta(X)^i[k] \in \mathbb{R}$ corresponds to the k -th index of $f_\theta(X)$ at pixel i . Let $N = W \times H$ be the number of pixels in an image.

Usual semantic segmentation frameworks, similarly to standard classifiers, apply a *softmax* function at the end of the neural network, obtaining per-pixel pseudo-probabilities for each class given by:

$$p_\theta(Y^i|X) = \frac{\exp(f_\theta(X)^i[Y^i])}{\sum_{k=1}^K \exp(f_\theta(X)^i[k])}. \quad (4.13)$$

Inspired from JEM, our goal is then to define two energy-based models, one for $p(X, Y)$ and another one for $p(X)$, such that $p(Y|X)$ matches the classical cross entropy computation.

Assuming that pixel classes are independent, given a certain image –which is an implicit assumption in semantic segmentation models trained with the usual cross-

entropy loss– we can express the probability $p_\theta(Y|X)$ as:

$$p_\theta(Y|X) = \prod_{i=1}^N p_\theta(Y^i|X) = \frac{\exp(\sum_i f_\theta(X)^i[Y^i])}{\prod_i \sum_k \exp(f_\theta(X)^i[k])}. \quad (4.14)$$

Taking this into account, we propose an energy based model for the joint distribution $p_\theta(X, Y)$ expressed as:

$$p_\theta(X, Y) = \frac{\prod_i \exp(f_\theta(X)^i[Y^i])}{Z(\theta)} = \frac{\exp(\sum_i f_\theta(X)^i[Y^i])}{Z(\theta)}. \quad (4.15)$$

Then $p_\theta(X)$ can be obtained by summing over all possible values of Y . However, this represents a problem, since this is a sum over *all possible* segmentation maps, and –in theory– there are K^N possible segmentation maps for an image:

$$p_\theta(X) = \sum_{\tilde{Y} \in \{1, \dots, K\}^N} \frac{\exp(\sum_i f_\theta(X)^i[\tilde{Y}^i])}{Z(\theta)}. \quad (4.16)$$

We can then derive $p_\theta(Y|X)$ from equations (4.15) and (4.16), which gives:

$$\begin{aligned} p_\theta(Y|X) &= \frac{p_\theta(X, Y)}{p_\theta(X)} = \frac{\exp(\sum_i f_\theta(X)^i[Y^i])}{\sum_{\tilde{Y} \in \{1, \dots, K\}^N} \exp(\sum_i f_\theta(X)^i[\tilde{Y}^i])} \\ &= \frac{\exp(\sum_i f_\theta(X)^i[Y^i])}{\sum_{\tilde{Y} \in \{1, \dots, K\}^N} \prod_i \exp(f_\theta(X)^i[\tilde{Y}^i])}. \end{aligned} \quad (4.17)$$

For this model to be an extension of JEM to semantic segmentation, we need to answer the following question: *Is the result in equation (4.17) equal to (4.14)?*

Or, in other words, is $\prod_i \sum_k \exp(f_\theta(X)^i[k]) = \sum_{\tilde{Y} \in \{1, \dots, K\}^N} \prod_i \exp(f_\theta(X)^i[\tilde{Y}^i])$?

The answer, given our assumptions, is yes. Remembering that $\sum_Y p_\theta(Y|X) = 1$, and since we assumed pixel-class independence, from equation (4.14) we have that:

$$\sum_{Y \in \{1, \dots, K\}^N} \prod_i p_\theta(Y^i|X) = \frac{\sum_{\tilde{Y} \in \{1, \dots, K\}^N} \prod_i \exp(f_\theta(X)^i[\tilde{Y}^i])}{\prod_i \sum_k \exp(f_\theta(X)^i[k])} = 1. \quad (4.18)$$

Therefore, we have extended the JEM approach to segmentation.

The joint energy function is then given by $E_\theta(X, Y) = -\sum_i f_\theta(X)^i[Y^i]$. And the energy function for the marginal distribution of X , from which we will sample, is then:

$$E_{\theta}(X) = -\log \left(\prod_i \sum_k \exp(f_{\theta}(X)^i[k]) \right) \quad (4.19)$$

As in JEM for classification, we can optimize the joint log-likelihood by writing it as: $p_{\theta}(X, Y) = p_{\theta}(X) + p_{\theta}(Y|X)$, where the second term is optimized as the usual cross-entropy in segmentation and the first term can be optimized using the same sampler as in JEM (SGLD, see Eq. (4.12)) and the energy given by Eq. (4.19).

Implementation of the SegJEM⁵ model raises some issues. JEM for classification is already difficult to train, due to the instability of gradient approximations; together with the intricacy of finding the right hyper-parameters for optimization and convergence. Semantic segmentation is a more complex task, therefore we can expect the optimization process to be at least equally arduous. Moreover, as mentioned in Section 4.4 an important limitation of current EBMs is training time –mostly due to MCMC sampling– which scales linearly with image size. Scenes for segmentation are usually large and, even if patch-based solutions might be applied, computational time will be a major constraint for this approach.

4.6 Conclusions

This chapter focused on the study of **semi-supervised learning from a generative point of view**. To this end, we have first defined what generative models are and briefly explained the main principles of different deep generative frameworks.

Despite some drawbacks, **energy-based models have several advantages** with respect to other generative models. They capture all the information about inputs only through a scalar value, *the energy*. Estimating the energy by the means of a neural network, enables to model complex distributions and makes EBMs very attractive for several applications, including generation, out-of-distribution detection, etc. Moreover, their simplicity allows for natural integration of label information into the model, by estimating a joint energy function $E(x, y)$, with very little changes on the neural network architecture to use, and no change on the optimization process.

In this context, we have considered a recent framework to train neural networks to jointly perform classification and generation of images and applied it to remote sensing data. By re-interpreting the outputs of a classification neural network, the **Joint**

5. JEM for segmentation

Energy-based Model (JEM) expresses the joint distribution of image-label pairs as an energy-based model. In practice, **it allows us to train a robust classifier and estimate the underlying distribution of data, simultaneously**. Moreover, this hybrid model is well suited and extends naturally to perform semi-supervised learning.

This seminal application of JEM to EO data led to several important conclusions. First, in small-scale datasets like EuroSAT, we observe that **JEM is a strong classifier** with performance on par with state-of-the-art methods. More interestingly, **in the semi-supervised setting when very few labeled examples are available, JEM is superior to a standard supervised network**, both in terms of classification scores and robustness (i.e. better calibrated). Second, with more realistic, large-scale datasets like So2Sat, JEM exhibits outstanding generalization properties, with better performance than usual classifiers in the supervised and semi-supervised settings. However, future work could focus on the integration in JEM of FixMatch mechanisms especially designed for semi-supervised learning, namely data augmentation techniques, pseudo-labeling or consistency regularization strategies. The challenge lies in realistically augmenting the data, and the distribution estimate given by JEM could be an asset here.

We have also demonstrated that JEM is able to correctly estimate the data distribution, allowing us to **generate faithful and diverse images**. Estimating the data distribution enables the model to detect out-of-distribution samples and thus to decide if it can be reliably used in a new domain. This gives JEM the ability to classify unseen zones with a confidence map based on the log-likelihood estimated by the model.

Despite the limitations and convergence issues of training energy-based models, we have shown through our experiments **several appealing applications in remote sensing** for this kind of hybrid discriminative-generative model, such as **semi-supervised learning, out-of-distribution detection or the generation of realistic new data**. It is a starting point to pave the way to tomorrow's real-life applications.

Finally, we have presented a **theoretical extension of this joint energy-based model to semantic segmentation**, SegJEM. However, the practical application of this extension is still at very early stages, thus it is left open as perspectives for future works.

CONCLUSION AND PERSPECTIVES ON FUTURE WORK

This work aims to make a contribution on the journey toward large-scale automated cartography. This is a difficult task because the Earth's surface is constantly changing due to several factors: seasons, human activity, natural disasters, climate change, among others. Even though today we have a plethora of data at our disposal –thanks to satellites and airborne campaigns–, it is not humanly possible to process and analyze effectively these data and to extract useful information in real-time. Therefore, the use of artificial intelligence emerges as a solution to achieve automatic analysis of EO imagery.

In recent years, deep learning techniques have been used and adapted by the remote sensing community, and have shown to be useful in various Earth observation applications. The main issue of these approaches is that to achieve their maximum potential, they need to be trained on large collections of labeled data. Gathering massive amounts of annotated data demands large efforts, time and resources, being infeasible in most practical cases. On the contrary, unlabeled data are abundant and easily available. Therefore, the development of algorithms that rely less on labels is a necessary step to leverage the insightful information contained on these unlabeled data.

Based on these considerations, this work has focused on the development of semi-supervised learning techniques for semantic segmentation and scene classification, toward large-scale cartography and EO data understanding. The idea is to fully exploit the large amounts of unlabeled data that are continuously gathered in Earth observation, and to integrate them into the learning algorithms, with the aim of developing robust and generic models.

Summary of contributions

Through our experiments and analysis, we have shown that there exist **several ways to integrate unlabeled data into deep learning models**, and each method comes with advantages and issues. However, the main take-home message is that we can leverage

completely unlabeled data to train algorithms with better performances and generalization capacities.

To this end, we started by **analyzing existing EO datasets** (see Chapter 2): do they represent real applications? **what are the desirable properties of a good dataset**, adapted to our objectives? Our goal is to achieve cartography at a large-scale⁶, therefore models need to generalize well across geographic locations. Moreover, labeled data are usually limited, while unlabeled data are easily accessible. Do existing datasets measure these situations?

We studied the generalization capacities of standard supervised semantic segmentation methods and we observed that they present generalization issues in large-scale settings, when labeled data are not diverse enough. However, we are aware that producing annotated data at a global-scale is impossible. Therefore, there is an opportunity for new learning paradigms to arise. In particular, this work focused on **semi-supervised learning to exploit the immensity of unlabeled data available**.

We constructed the **MiniFrance suite**, a large-scale dataset especially designed for semi-supervised semantic segmentation in Earth observation. Because of its unique qualities and design, we hope it will encourage and push the research limits on semi-supervised semantic segmentation in the field. In this context, **MiniFrance is part of the renowned IEEE GRSS Data Fusion Contest 2022**. This year's theme is semi-supervised learning, and the DFC-MF22 dataset is an extended version of MiniFrance to new modalities. Therefore, we have contributed to the organization of this important competition that every year gathers researchers from all over the world.

We also presented **tools for representativeness data analysis** in the context of multi-location datasets. They allow practitioners to assess their data prior to any experiment. We illustrate the use of these tools on the MiniFrance data, defining a suitable partition for semi-supervised learning.

Further, in Chapter 3 we delved into the development of semi-supervised techniques from a **discriminative perspective**, in particular, we studied two kinds of models: **multi-task learning and consistency regularization-based approaches**. Multi-task learning is a straightforward⁷ way to integrate unlabeled data into the training loop of standard supervised networks –obtaining a semi-supervised network–, by simply adding an unsupervised, secondary task. Thereby, we introduced neural networks to tackle

6. even global, if possible.

7. and maybe the most straightforward.

semi-supervised semantic segmentation from a multi-task perspective –in particular, BerundaNet–, training the network to simultaneously perform supervised segmentation and an unsupervised, auxiliary task. Our experiments on three public benchmarks, including MiniFrance, –where **we explored different unsupervised tasks and corresponding loss functions**– showed the benefits we can obtain by integrating unlabeled data into the learning process. Still, the multi-task setting leaves some open questions: how to choose the right auxiliary task? which is the best loss function to optimize? do we always get better results by using extra unlabeled data? The answer to them is not simple, and one has to settle for empirical results.

The second family of discriminative methods that we explored in Chapter 3 is based on the principle of consistency training. These methods rely on the assumption that a model should output similar predictions for semantically similar inputs. We presented a theoretical framework for this kind of methods and study the transferability of **Fix-Match**, the current state-of-the-art for semi-supervised classification in computer vision, to remote sensing data. From our experiments we observed the potential of this method to tackle semi-supervised scene classification in remote sensing, even when there exists a domain shift between training and test data.

Our study on semi-supervised learning continues in Chapter 4 by exploring the **semi-supervised problem from a generative perspective**. As our analysis in Chapter 2 has shown, data distribution contains valuable information that is not directly exploited by discriminative models (Chapter 3). Therefore, estimating the data distribution of EO images through generative models is a suitable solution to integrate this unexploited knowledge, especially when large amounts of (unlabeled) data are available. We investigate **energy-based models for various Earth observation applications**. EBMs have several advantages over other generative models. In particular, they allow us to naturally integrate label information into the model, without significant changes of the learning process. We performed experiments with **JEM** –a recent framework for joint classification and generation– on remote sensing data. This hybrid model can be naturally extended to a semi-supervised setting. Our experiments on several public datasets for scene classification demonstrate that **JEM is transferable to the EO domain, and that it is a strong classifier**, with performance on par to state-of-the-art methods. Moreover, **in semi-supervised settings it shows considerable improvements with respect to a purely supervised method** trained on the same number of labeled samples, with good generalization capacities with respect to domain shifts.

Furthermore, **JEM is able to estimate the data distribution which allows the model to generate realistic and diverse new data**. Additionally, thanks to the traceability of the likelihood function, the model is able to detect out-of-distribution samples and thus to decide if it can be reliably used in a new, unseen domain.

Despite the issues regarding training of EBMs, we observed that these models have numerous **appealing applications for remote sensing data**. We believe that there is still much progress to make to improve and optimize EBMs' training and they can lead to build real-life applications.

Overall, in this manuscript we have explored semi-supervised learning techniques to progress toward large-scale automated cartography and EO data understanding. We have explored several aspects of semi-supervised semantic segmentation in Earth observation, proposing a new benchmark –MiniFrance– and different methods to achieve semi-supervised classification and segmentation, including BerundaNet, FixMatch for EO and JEM for EO. We have performed an analysis of these methods, highlighting their advantages and also their limitations. However, the problem of semi-supervised learning and its applications is still open and several perspectives for future work arise.

Perspectives for future work

Short-term projects

1. FixMatch extension for semantic segmentation. Our experiments in Section 3.3 showed that classification methods based on consistency regularization, like FixMatch, can have impressive results in semi-supervised settings when very few labeled data are available. A natural following step is try to extend these ideas to dense, pixel-wise classification, and to land use/land cover maps. The challenge is then to find the right data augmentations to apply to data, adapted to segmentation and to the EO domain. Some works in this direction have appeared recently [83, 84].

2. Can we integrate consistency regularization into a generative framework? Experiments in Section 3.3 demonstrated the effectiveness of consistency regularization and other common techniques for semi-supervised learning, *e.g.* pseudo-labeling, for semi-supervised scene classification in EO data. Moreover, experiments in Chapter 4 showed multiple applications of EBMs with high interest for the remote sensing community.

Then, a natural question arises: can we combine these approaches and get the best of both worlds? The idea is to integrate the elements that make the success of FixMatch into the hybrid JEM framework to get a robust semi-supervised classifier and generative model. The challenge lies in designing realistic data transformations that allow us to train a robust classifier and, simultaneously, a good generative model.

3. JEM extension for semantic segmentation. In section 4.5 we proposed an extension of JEM to semantic segmentation and demonstrated its theoretical feasibility. However, the proposed model was not tested in practical applications. The challenge here is mostly to solve practical issues related to training and convergence of the model.

Long-term projects

1. Noisy labels. A recurrent issue when dealing with large-scale data, and in particular in semantic segmentation, is how to treat noisy annotations. In general, gathering pixel-wise annotations for large-scale data is extremely difficult (if not impossible), and the use of semi-automatic processes, which are not 100% accurate (this is the case for label information of MiniFrance, Section 2.3), is a common way to obtain labels. In this work, we ignored the presence of these noisy labels, but it is an important topic of research when working at a large, global scale [111].

2. Domain adaptation. Another interesting research topic that we mentioned during this work, without tackling it directly, is domain adaptation [215]: trying to train domain-agnostic models that generalize equally to any geographic location. Indeed, unsupervised domain adaptation (UDA) methods can be considered as a special case of semi-supervised learning: on both settings we have labeled and unlabeled data, but UDA makes the additional assumption that both sets come from different domains (source domain labeled data and target domain unlabeled data). The objective is then to classify the target images, based on knowledge (labeled data) from the source domain. Given the similarities between both problems, it would be interesting to develop methods that integrate semi-supervised techniques to achieve or improve domain adaptation methods.

3. Open-set world. Domain shifts are not the only issue we can find when trying to apply models to new geographic locations. Indeed, classes may change between different

areas on Earth: how would a model trained on European agricultural images classify aerial images of the Atacama desert or the Uyuni salt flat? Open-set algorithms [216] refer to the methods that try to solve this general problem, where test data may come with additional classes that have not been seen during training. How to deal with them? How to detect them? Maybe semi-supervised learning can help to make more progress in this direction, introducing data variability through unlabeled samples.

4. Integrate domain knowledge into our models. In this work, we investigated semi-supervised learning in Earth observation applications. However, we focused on the image perspective, feeding our models only with optical data. A next step would be to adapt our methods to take into account all the information that is available in EO data. For instance, can we integrate geographical position information into our models? Can we incorporate the physics governing the Earth system into our models? In general, we can leverage unlabeled data to take into account the recommendations of Reichstein et al. [2].

Solving –or at least making progress on– these last four topics would transform the analysis of EO data as we know it today. Indeed, we would be able to build big models capable of integrating all new observations, regardless of their source. We could feed these models with unlabeled data or with noisy labeled data, they would be able to detect noise on annotations and self-correct them. Moreover, they would integrate physical constraints and specific domain knowledge, making them a more robust description of reality. These kind of models could be applied everywhere, enabling real land use classification at a global-scale, taking into account domain shifts and even new land use classes. In that scenario, we would be able to detect anomalies and changes accurately and in real-time, allowing for rapid response to extreme events.

BIBLIOGRAPHY

- [1] G. Camps-Valls, D. Tuia, L. Gómez-Chova, S. Jiménez, and J. Malo, « Remote sensing image processing », *Synthesis Lectures on Image, Video, and Multimedia Processing*, vol. 5, 1, pp. 1–192, 2011 (cit. on pp. 11, 22).
- [2] M. Reichstein, G. Camps-Valls, B. Stevens, M. Jung, J. Denzler, N. Carvalhais, et al., « Deep learning and process understanding for data-driven earth system science », *Nature*, vol. 566, 7743, pp. 195–204, 2019 (cit. on pp. 11, 12, 23, 59, 166).
- [3] A. Agapiou, « Remote sensing heritage in a petabyte-scale: satellite data and heritage earth engine© applications », *International Journal of Digital Earth*, vol. 10, 1, pp. 85–102, 2017 (cit. on pp. 11, 23).
- [4] N. Audebert, « Classification de données massives de télédétection », Ph.D. dissertation, Université Bretagne Sud, 2018 (cit. on pp. 12, 23).
- [5] R. Caye Daudt, « Convolutional neural networks for change analysis in earth observation images with noisy labels and domain shifts », Ph.D. dissertation, Institut polytechnique de Paris, 2020 (cit. on pp. 12, 23).
- [6] J. Castillo-Navarro, N. Audebert, A. Boulch, B. Le Saux, and S. Lefèvre, « What data are needed for semantic segmentation in Earth observation? », in *2019 Joint Urban Remote Sensing Event (JURSE)*, IEEE, 2019.
- [7] J. Castillo-Navarro, B. Le Saux, A. Boulch, and S. Lefèvre, « Réseaux de neurones semi-supervisés pour la segmentation sémantique en télédétection », in *Colloque GRETSI*, 2019.
- [8] J. Castillo-Navarro, B. Le Saux, A. Boulch, and S. Lefèvre, « On auxiliary losses for semi-supervised semantic segmentation », in *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery Workshops (ECML-PKDD W) - MACLEAN*, 2020.
- [9] J. Castillo-Navarro, B. Le Saux, A. Boulch, and S. Lefèvre, « Energy-based models in Earth observation: from generation to semi-supervised learning », *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, 2022.

-
- [10] J. Castillo-Navarro, B. Le Saux, A. Boulch, and S. Lefèvre, « Classification and generation of Earth observation images using a joint energy-based model », in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2021.
- [11] J. Castillo-Navarro, B. Le Saux, A. Boulch, and S. Lefèvre, « Energy-based models for Earth observation applications », in *Proceedings of the International Conference on Learning Representations - Energy Based Models Workshop - (ICLR-W)*, 2021.
- [12] R. Hänsch, C. Persello, G. Vivone, J. Castillo-Navarro, A. Boulch, S. Lefèvre, and B. Le Saux, « 2022 IEEE GRSS Data fusion contest: semi-supervised learning [technical committees] », to appear in *IEEE Geoscience and Remote Sensing Magazine*, 2022.
- [13] R. K. Runtz, S. Phinn, Z. Xie, O. Venter, and J. E. Watson, « Opportunities for big data in conservation and sustainability », *Nature Communications*, vol. 11, 1, pp. 1–4, 2020 (cit. on p. 22).
- [14] T. Inomata, D. Triadan, V. A. V. López, J. C. Fernandez-Diaz, T. Omori, M. B. M. Bauer, M. G. Hernández, T. Beach, C. Cagnato, K. Aoyama, *et al.*, « Monumental architecture at aguada fénix and the rise of maya civilization », *Nature*, vol. 582, 7813, pp. 530–533, 2020 (cit. on p. 22).
- [15] P. O. Gray, *Psychology, 5th Edition*, 5th. 2006 (cit. on p. 32).
- [16] S. Hao, Y. Zhou, and Y. Guo, « A brief survey on semantic segmentation with deep learning », *Neurocomputing*, vol. 406, pp. 302–321, 2020 (cit. on pp. 32, 34).
- [17] D. R. Martin, C. C. Fowlkes, and J. Malik, « Learning to detect natural image boundaries using local brightness, color, and texture cues », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, 5, pp. 530–549, 2004 (cit. on p. 33).
- [18] H. Zhu, F. Meng, J. Cai, and S. Lu, « Beyond pixels: a comprehensive survey from bottom-up to semantic image segmentation and cosegmentation », *Journal of Visual Communication and Image Representation*, vol. 34, pp. 12–27, 2016 (cit. on p. 34).

-
- [19] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, P. Martinez-Gonzalez, and J. Garcia-Rodriguez, « A survey on deep learning techniques for image and video semantic segmentation », *Applied Soft Computing*, vol. 70, pp. 41–65, 2018 (cit. on p. 34).
- [20] X. Liu, Z. Deng, and Y. Yang, « Recent progress in semantic image segmentation », *Artificial Intelligence Review*, vol. 52, 2, pp. 1089–1106, 2019 (cit. on p. 34).
- [21] S. Minaee, Y. Y. Boykov, F. Porikli, A. J. Plaza, N. Kehtarnavaz, and D. Terzopoulos, « Image segmentation using deep learning: a survey », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021 (cit. on pp. 34, 49).
- [22] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, « Learning hierarchical features for scene labeling », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, 8, pp. 1915–1929, 2012 (cit. on p. 34).
- [23] J. Long, E. Shelhamer, and T. Darrell, « Fully convolutional networks for semantic segmentation », in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015 (cit. on pp. 34, 48).
- [24] T. M. Mitchell, *The discipline of machine learning*. 2006, vol. 9 (cit. on p. 36).
- [25] O. Chapelle, B. Schölkopf, and A. Zien, *Semi-Supervised Learning*. 2006 (cit. on pp. 37, 38).
- [26] Z.-H. Zhou, « A brief introduction to weakly supervised learning », *National science review*, vol. 5, 1, pp. 44–53, 2018 (cit. on p. 37).
- [27] L. Jing and Y. Tian, « Self-supervised visual feature learning with deep neural networks: a survey », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, 11, pp. 4037–4058, 2021 (cit. on p. 37).
- [28] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, « A simple framework for contrastive learning of visual representations », in *Proceedings of the International Conference on Machine Learning (ICML)*, 2020 (cit. on pp. 38, 102).
- [29] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, « Momentum contrast for unsupervised visual representation learning », in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020 (cit. on pp. 38, 102).
- [30] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. 2016, <http://www.deeplearningbook.org> (cit. on pp. 41, 99, 133).

-
- [31] W. S. McCulloch and W. Pitts, « A logical calculus of the ideas immanent in nervous activity », *The bulletin of mathematical biophysics*, vol. 5, 4, pp. 115–133, 1943 (cit. on p. 41).
- [32] F. Rosenblatt, « The perceptron: a probabilistic model for information storage and organization in the brain. », *Psychological review*, vol. 65, 6, p. 386, 1958 (cit. on p. 41).
- [33] H. J. Kelley, « Gradient theory of optimal flight paths », *Ars Journal*, vol. 30, 10, pp. 947–954, 1960 (cit. on p. 41).
- [34] S. Dreyfus, « The numerical solution of variational problems », *Journal of Mathematical Analysis and Applications*, vol. 5, 1, pp. 30–45, 1962 (cit. on p. 41).
- [35] M. L. Minsky and Papert, *Perceptrons*. 1969 (cit. on p. 42).
- [36] S. Linnainmaa, « The representation of the cumulative rounding error of an algorithm as a taylor expansion of the local rounding errors », *Master's Thesis (in Finnish)*, *Univ. Helsinki*, pp. 6–7, 1970 (cit. on p. 42).
- [37] A. G. Ivakhnenko, « Polynomial theory of complex systems », *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-1, 4, pp. 364–378, 1971 (cit. on p. 42).
- [38] K. Fukushima and S. Miyake, « Neocognitron: a self-organizing neural network model for a mechanism of visual pattern recognition », in *Competition and Cooperation in Neural Nets*, 1982, pp. 267–285 (cit. on p. 42).
- [39] P. Werbos, « Beyond regression: new tools for prediction and analysis in the behavioral sciences », *Ph. D. dissertation, Harvard University*, 1974 (cit. on p. 42).
- [40] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, « Learning representations by back-propagating errors », *Nature*, vol. 323, 6088, pp. 533–536, 1986 (cit. on p. 42).
- [41] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, « Backpropagation applied to handwritten zip code recognition », *Neural Computation*, vol. 1, 4, pp. 541–551, 1989 (cit. on pp. 42, 46).
- [42] G. Cybenko, « Approximation by superpositions of a sigmoidal function », *Mathematics of Control, Signals and Systems*, vol. 2, 4, pp. 303–314, 1989 (cit. on p. 42).

-
- [43] D. Steinkraus, I. Buck, and P. Simard, « Using GPUs for machine learning algorithms », in *Proceedings of the IEEE International Conference on Document Analysis and Recognition (ICDAR)*, 2005 (cit. on p. 43).
- [44] K. Chellapilla, S. Puri, and P. Simard, « High performance convolutional neural networks for document processing », in *Proceedings of the International Workshop on Frontiers in Handwriting Recognition*, Suvisoft, 2006 (cit. on p. 43).
- [45] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, « ImageNet: A large-scale hierarchical image database », in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009 (cit. on pp. 43, 59).
- [46] A. Krizhevsky, I. Sutskever, and G. Hinton, « ImageNet classification with deep convolutional neural networks », in *Advances in Neural Information Processing Systems (NeurIPS)*, 2012 (cit. on pp. 43, 46).
- [47] Y. LeCun, Y. Bengio, and G. Hinton, « Deep learning », *Nature*, vol. 521, 7553, pp. 436–444, 2015 (cit. on p. 43).
- [48] M. Leshno, V. Y. Lin, A. Pinkus, and S. Schocken, « Multilayer feedforward networks with a nonpolynomial activation function can approximate any function », *Neural Networks*, vol. 6, 6, pp. 861–867, 1993 (cit. on p. 45).
- [49] S. Hochreiter, « Untersuchungen zu dynamischen neuronalen netzen », *Diploma, Technische Universität München*, vol. 91, 1, 1991 (cit. on p. 45).
- [50] X. Glorot, A. Bordes, and Y. Bengio, « Deep sparse rectifier neural networks », in *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2011 (cit. on p. 45).
- [51] A. Apicella, F. Donnarumma, F. Isgrò, and R. Prevete, « A survey on modern trainable activation functions », *Neural Networks*, vol. 138, pp. 14–32, 2021 (cit. on p. 45).
- [52] Y. LeCun, L. Bottou, and Y. Bengio, « Reading checks with multilayer graph transformer networks », in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 1997 (cit. on p. 46).
- [53] V. Dumoulin and F. Visin, « A guide to convolution arithmetic for deep learning », *arXiv preprint arXiv:1603.07285*, 2016 (cit. on p. 48).

-
- [54] J. J. Zhao, M. Mathieu, R. Goroshin, and Y. LeCun, « Stacked what-where auto-encoders », in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2015 (cit. on p. 49).
- [55] V. Badrinarayanan, A. Kendall, and R. Cipolla, « SegNet: a deep convolutional encoder-decoder architecture for image segmentation », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, 12, pp. 2481–2495, 2017 (cit. on pp. 49, 63, 101).
- [56] H. Noh, S. Hong, and B. Han, « Learning deconvolution network for semantic segmentation », in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015 (cit. on p. 49).
- [57] O. Ronneberger, P. Fischer, and T. Brox, « U-Net: convolutional networks for biomedical image segmentation », in *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2015 (cit. on pp. 49, 50, 101).
- [58] K. He, X. Zhang, S. Ren, and J. Sun, « Deep residual learning for image recognition », in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016 (cit. on pp. 49, 78).
- [59] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, « DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, 4, pp. 834–848, 2017 (cit. on p. 49).
- [60] A. Chaurasia and E. Culurciello, « LinkNet: exploiting encoder representations for efficient semantic segmentation », in *Proceedings of the IEEE International Conference on Visual Communications and Image Processing (VCIP)*, 2017 (cit. on p. 49).
- [61] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, « Pyramid scene parsing network », in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017 (cit. on p. 49).
- [62] A. Oliver, A. Odena, C. A. Raffel, E. D. Cubuk, and I. Goodfellow, « Realistic evaluation of deep semi-supervised learning algorithms », in *Advances in Neural Information Processing Systems (NeurIPS)*, 2018 (cit. on p. 50).
- [63] Y. Ouali, C. Hudelot, and M. Tami, « An overview of deep semi-supervised learning », *arXiv preprint arXiv:2006.05278*, 2020 (cit. on p. 51).

-
- [64] S. Laine and T. Aila, « Temporal ensembling for semi-supervised learning », in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017 (cit. on pp. 51, 120).
- [65] A. Tarvainen and H. Valpola, « Mean teachers are better role models: weight-averaged consistency targets improve semi-supervised deep learning results », in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017 (cit. on pp. 51, 120).
- [66] T. Miyato, S.-i. Maeda, M. Koyama, and S. Ishii, « Virtual adversarial training: a regularization method for supervised and semi-supervised learning », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, 8, pp. 1979–1993, 2018 (cit. on pp. 51, 120).
- [67] D.-H. Lee, « Pseudo-Label : The Simple and Efficient Semi-Supervised Learning Method for Deep Neural Networks », in *Proceedings of the International Conference on Machine Learning - Workshop on Challenges in Representation learning (ICML-W)*, 2013 (cit. on pp. 51, 120, 123).
- [68] D. P. Kingma, S. Mohamed, D. J. Rezende, and M. Welling, « Semi-supervised learning with deep generative models », in *Advances in Neural Information Processing Systems (NeurIPS)*, 2014 (cit. on pp. 51, 135).
- [69] A. Odena, « Semi-supervised learning with generative adversarial networks », *arXiv preprint arXiv:1606.01583*, 2016 (cit. on p. 51).
- [70] Z. Dai, Z. Yang, F. Yang, W. W. Cohen, and R. R. Salakhutdinov, « Good semi-supervised learning that requires a bad GAN », in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017 (cit. on p. 51).
- [71] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C. A. Raffel, « MixMatch: A holistic approach to semi-supervised learning », in *Advances in Neural Information Processing Systems (NeurIPS)*, 2019 (cit. on pp. 51, 120, 124).
- [72] D. Berthelot, N. Carlini, E. D. Cubuk, A. Kurakin, K. Sohn, H. Zhang, and C. Raffel, « ReMixMatch: Semi-supervised learning with distribution matching and augmentation anchoring. », in *Proceedings of the International Conference on Machine Learning (ICML)*, 2020 (cit. on p. 51).

-
- [73] K. Sohn, D. Berthelot, C.-L. Li, Z. Zhang, N. Carlini, E. D. Cubuk, A. Kurakin, H. Zhang, and C. Raffel, « FixMatch: Simplifying semi-supervised learning with consistency and confidence », in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020 (cit. on pp. 51, 120, 123, 128, 147).
- [74] L. Maggiolo, D. Marcos, G. Moser, and D. Tuia, « Improving maps from CNNs trained with sparse, scribbled ground truths using fully connected CRFs », in *IEEE International Symposium on Geoscience and Remote Sensing (IGARSS)*, 2018 (cit. on p. 51).
- [75] T. Durand, T. Mordan, N. Thome, and M. Cord, « WILDCAT: weakly supervised learning of deep ConvNets for image classification, pointwise localization and segmentation », in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017 (cit. on p. 51).
- [76] A. Khoreva, R. Benenson, J. H. Hosang, M. Hein, and B. Schiele, « Simple does it: weakly supervised instance and semantic segmentation », in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017 (cit. on p. 51).
- [77] G. Papandreou, L.-C. Chen, K. P. Murphy, and A. L. Yuille, « Weakly-and semi-supervised learning of a deep convolutional network for semantic image segmentation », in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015 (cit. on p. 51).
- [78] Z. Chen, R. Zhang, G. Zhang, Z. Ma, and T. Lei, « Digging into pseudo label: a low-budget approach for semi-supervised semantic segmentation », *IEEE Access*, vol. 8, pp. 41 830–41 837, 2020 (cit. on p. 52).
- [79] N. Souly, C. Spampinato, and M. Shah, « Semi-supervised semantic segmentation using generative adversarial network », in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017 (cit. on p. 52).
- [80] W.-C. Hung, Y.-H. Tsai, Y.-T. Liou, Y.-Y. Lin, and M.-H. Yang, « Adversarial learning for semi-supervised semantic segmentation », *Proceedings of the British Machine Vision Conference (BMVC)*, 2018 (cit. on p. 52).
- [81] T. Kalluri, G. Varma, M. Chandraker, and C. Jawahar, « Universal semi-supervised semantic segmentation », in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019 (cit. on p. 52).

-
- [82] Y. Ouali, C. Hudelot, and M. Tami, « Semi-supervised semantic segmentation with cross-consistency training », in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020 (cit. on p. 52).
- [83] Y. Zou, Z. Zhang, H. Zhang, C.-L. Li, X. Bian, J.-B. Huang, and T. Pfister, « Pseudoseg: designing pseudo labels for semantic segmentation », in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021 (cit. on pp. 52, 164).
- [84] X. Chen, Y. Yuan, G. Zeng, and J. Wang, « Semi-supervised semantic segmentation with cross pseudo supervision », in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021 (cit. on pp. 52, 164).
- [85] D. Li, J. Yang, K. Kreis, A. Torralba, and S. Fidler, « Semantic segmentation with generative models: semi-supervised learning and strong out-of-domain generalization », in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021 (cit. on p. 52).
- [86] X. X. Zhu, D. Tuia, L. Mou, G.-S. Xia, L. Zhang, F. Xu, and F. Fraundorfer, « Deep learning in remote sensing: a comprehensive review and list of resources », *IEEE Geoscience and Remote Sensing Magazine*, vol. 5, 4, pp. 8–36, 2017 (cit. on pp. 52, 53).
- [87] L. Ma, Y. Liu, X. Zhang, Y. Ye, G. Yin, and B. A. Johnson, « Deep learning in remote sensing applications: a meta-analysis and review », *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 152, pp. 166–177, 2019 (cit. on p. 53).
- [88] R. C. Daudt, B. Le Saux, and A. Boulch, « Fully convolutional siamese networks for change detection », in *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, 2018 (cit. on p. 53).
- [89] V. Ferraris, N. Dobigeon, Y. Cavalcanti, T. Oberlin, and M. Chabert, « Unsupervised change detection for multimodal remote sensing images via coupled dictionary learning and sparse coding », in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2020 (cit. on p. 53).
- [90] J. Vargas-Munoz, S. Lobry, A. Falcao, and D. Tuia, « Correcting rural building annotations in OpenStreetMap using convolutional neural networks », *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 10, pp. 283–293, 2019 (cit. on p. 53).

-
- [91] G. Sumbul, M. Charfuelan, B. Demir, and V. Markl, « BigEarthNet: a large-scale benchmark archive for remote sensing image understanding », in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2019 (cit. on pp. 53, 60, 144, 152).
- [92] M. Carvalho, B. Le Saux, P. Trouvé-Peloux, A. Almansa, and F. Champagnat, « Multi-task learning of height and semantics from aerial images », *IEEE Geoscience and Remote Sensing Letters*, vol. 17, 8, pp. 1391–1395, (cit. on pp. 53, 101).
- [93] N. Audebert, B. Le Saux, and S. Lefèvre, « Beyond RGB: very high resolution urban remote sensing with multimodal deep networks », *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 140, pp. 20–32, 2018 (cit. on pp. 53, 62).
- [94] G. Camps-Valls, D. Tuia, X. X. Zhu, and M. Reichstein, *Deep learning for the Earth Sciences: A comprehensive approach to remote sensing, climate science and geosciences*. 2021 (cit. on pp. 53, 59).
- [95] D. Tuia, R. Roscher, J. D. Wegner, N. Jacobs, X. Zhu, and G. Camps-Valls, « Toward a collective agenda on ai for earth science data analysis », *IEEE Geoscience and Remote Sensing Magazine*, vol. 9, 2, pp. 88–104, 2021 (cit. on p. 53).
- [96] A. Lagrange, B. Le Saux, A. Beaupère, A. Boulch, A. Chan-Hon-Tong, S. Herbin, H. Randrianarivo, and M. Ferecatu, « Benchmarking classification of earth-observation data: from learning explicit features to convolutional networks », in *IEEE Int. Geoscience and Remote Sensing Symposium (IGARSS)*, Milan, Italy, 2015 (cit. on p. 53).
- [97] J. Sherrah, « Fully convolutional networks for dense semantic labelling of high-resolution aerial imagery », *arXiv preprint arXiv:1606.02585*, 2016 (cit. on p. 54).
- [98] N. Audebert, B. Le Saux, and S. Lefèvre, « Semantic segmentation of Earth observation data using multimodal and multi-scale deep networks », in *Proceedings of the Asian Conference on Computer Vision (ACCV)*, 2016 (cit. on p. 54).
- [99] M. Kampffmeyer, A.-B. Salberg, and R. Jenssen, « Semantic segmentation of small objects and modeling of uncertainty in urban remote sensing images using deep convolutional neural networks », in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPR-W)*, 2016 (cit. on p. 54).

-
- [100] M. Volpi and D. Tuia, « Dense semantic labeling of subdecimeter resolution images with convolutional neural networks », *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, 2, pp. 881–893, 2017 (cit. on p. 54).
- [101] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez, « Convolutional neural networks for large-scale remote-sensing image classification », *IEEE Transactions on geoscience and remote sensing*, vol. 55, 2, pp. 645–657, 2017 (cit. on p. 54).
- [102] R. Caye Daudt, B. Le Saux, A. Boulch, and Y. Gousseau, « Multitask learning for large-scale semantic change detection », *Computer Vision and Image Understanding*, vol. 187, p. 102783, 2019 (cit. on p. 54).
- [103] G. Camps-Valls, T. V. B. Marsheva, and D. Zhou, « Semi-supervised graph-based hyperspectral image classification », *IEEE Transactions on Geoscience and Remote Sensing*, vol. 45, 10, pp. 3044–3054, 2007 (cit. on p. 54).
- [104] D. Hong, N. Yokoya, J. Chanussot, J. Xu, and X. X. Zhu, « Learning to propagate labels on graphs: an iterative multitask regression framework for semi-supervised hyperspectral dimensionality reduction », *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 158, pp. 35–49, 2019 (cit. on p. 54).
- [105] D. Tuia, M. Volpi, M. Trolliet, and G. Camps-Valls, « Semisupervised manifold alignment of multimodal remote sensing images », *IEEE Transactions on Geoscience and Remote Sensing*, vol. 52, 12, pp. 7708–7720, 2014 (cit. on p. 54).
- [106] B. Zhao, J. R. Sveinsson, M. O. Ulfarsson, and J. Chanussot, « Semi-supervised mixtures of factor analyzers and deep mixtures of factor analyzers dimensionality reduction algorithms for hyperspectral images classification », in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2019 (cit. on p. 54).
- [107] W. Han, R. Feng, L. Wang, and Y. Cheng, « A semi-supervised generative framework with deep learning features for high-resolution remote sensing image scene classification », *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 145, pp. 23–43, 2018 (cit. on p. 54).
- [108] R. Fan, R. Feng, L. Wang, J. Yan, and X. Zhang, « Semi-MCNN: a semisupervised multi-CNN ensemble learning method for urban land cover classification using submeter HRRS images », *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 4973–4987, 2020 (cit. on p. 54).

-
- [109] D. Hong, N. Yokoya, G.-S. Xia, J. Chanussot, and X. X. Zhu, « X-ModalNet: A semi-supervised deep cross-modal network for classification of remote sensing data », *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 167, pp. 12–23, 2020 (cit. on p. 54).
- [110] X.-Y. Tong, G.-S. Xia, Q. Lu, H. Shen, S. Li, S. You, and L. Zhang, « Land-cover classification with high-resolution remote sensing images using transferable deep models », *Remote Sensing of Environment*, vol. 237, p. 111 322, 2020 (cit. on p. 54).
- [111] R. C. Daudt, B. Le Saux, A. Boulch, and Y. Gousseau, « Weakly supervised change detection using guided anisotropic diffusion », *Machine Learning*, 2021 (cit. on pp. 54, 165).
- [112] E. Protopapadakis, A. Doulamis, N. Doulamis, and E. Maltezos, « Stacked autoencoders driven by semi-supervised learning for building extraction from near infrared remote sensing imagery », *Remote Sensing*, vol. 13, 3, p. 371, 2021 (cit. on p. 54).
- [113] R. Zhu, L. Yan, N. Mo, and Y. Liu, « Semi-supervised center-based discriminative adversarial learning for cross-domain scene-level land-cover classification of aerial images », *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 155, pp. 72–89, 2019 (cit. on p. 54).
- [114] X. Cao, J. Yao, Z. Xu, and D. Meng, « Hyperspectral image classification with convolutional neural network and active learning », *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, 7, pp. 4604–4616, 2020 (cit. on p. 54).
- [115] F. M. Riese, S. Keller, and S. Hinz, « Supervised and semi-supervised self-organizing maps for regression and classification focusing on hyperspectral data », *Remote Sensing*, vol. 12, 1, p. 7, 2020 (cit. on p. 54).
- [116] K. Zhang and H. Yang, « Semi-supervised multi-spectral land cover classification with multi-attention and adaptive kernel », in *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, 2020 (cit. on p. 55).
- [117] Y. Zhan, D. Hu, Y. Wang, and X. Yu, « Semisupervised hyperspectral image classification based on generative adversarial networks », *IEEE Geoscience and Remote Sensing Letters*, vol. 15, 2, pp. 212–216, 2017 (cit. on p. 55).

-
- [118] D. Guo, Y. Xia, and X. Luo, « GAN-based semisupervised scene classification of remote sensing image », *IEEE Geoscience and Remote Sensing Letters*, vol. 18, 12, pp. 2067–2071, 2021 (cit. on p. 55).
- [119] S. Roy, E. Sangineto, N. Sebe, and B. Demir, « Semantic-fusion GANs for semi-supervised satellite image classification », in *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, 2018 (cit. on p. 55).
- [120] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, « Microsoft COCO: Common Objects in Context », in *Proceedings of European Conference on Computer Vision (ECCV)*, 2014 (cit. on p. 59).
- [121] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, « The Cityscapes dataset for semantic urban scene understanding », in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016 (cit. on p. 59).
- [122] M. Schmitt, S. A. Ahmadi, and R. Hänsch, « There is no data like more data—current status of machine learning datasets in remote sensing », in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2021 (cit. on p. 59).
- [123] L. Alparone, L. Wald, J. Chanussot, C. Thomas, P. Gamba, and L. M. Bruce, « Comparison of pansharpening algorithms: outcome of the 2006 grs-s data fusion contest », *IEEE Transactions on Geoscience and Remote Sensing*, vol. 45, 10, pp. 3012–3021, 2007 (cit. on p. 59).
- [124] P. Helber, B. Bischke, A. Dengel, and D. Borth, « EuroSAT: A novel dataset and deep learning benchmark for land use and land cover classification », *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 12, 7, pp. 2217–2226, 2019 (cit. on pp. 60, 127, 143, 145, 152).
- [125] X. X. Zhu, J. Hu, C. Qiu, Y. Shi, J. Kang, L. Mou, H. Bagheri, M. Haberle, Y. Hua, R. Huang, L. Hughes, H. Li, Y. Sun, G. Zhang, S. Han, M. Schmitt, and Y. Wang, « So2Sat LCZ42: a benchmark data set for the classification of global local climate zones [software and data sets] », *IEEE Geoscience and Remote Sensing Magazine*, vol. 8, 3, pp. 76–89, 2020 (cit. on pp. 60, 127, 143).
- [126] G.-S. Xia, J. Hu, F. Hu, B. Shi, X. Bai, Y. Zhong, L. Zhang, and X. Lu, « AID: a benchmark data set for performance evaluation of aerial scene classification »,

-
- IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, 7, pp. 3965–3981, 2017 (cit. on pp. 60, 143).
- [127] Y. Yang and S. Newsam, « Bag-of-visual-words and spatial extensions for land-use classification », in *Proceedings of the SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 2010 (cit. on pp. 60, 143).
- [128] L. Kondmann, A. Toker, M. Rußwurm, A. Camero, D. Peressuti, G. Milcinski, P.-P. Mathieu, N. Longépé, T. Davis, G. Marchisio, *et al.*, « Denethor: the dynamic earthnet dataset for harmonized, inter-operable, analysis-ready, daily crop monitoring from space », in *Advances in Neural Information and Processing Systems (NeurIPS) - Datasets and Benchmarks Track*, 2021 (cit. on p. 60).
- [129] G.-S. Xia, X. Bai, J. Ding, Z. Zhu, S. Belongie, J. Luo, M. Datcu, M. Pelillo, and L. Zhang, « DOTA: a large-scale dataset for object detection in aerial images », in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018 (cit. on p. 60).
- [130] D. Lam, R. Kuzma, K. McGee, S. Dooley, M. Laielli, M. Klaric, Y. Bulatov, and B. McCord, « xView: objects in context in overhead imagery », *arXiv preprint arXiv:1802.07856*, 2018 (cit. on p. 60).
- [131] R. Gupta, B. Goodman, N. Patel, R. Hosfelt, S. Sajejev, E. Heim, J. Doshi, K. Lucas, H. Choset, and M. Gaston, « Creating xBD: A Dataset for Assessing Building Damage from Satellite Imagery », in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition - Workshop Computer Vision for Global Challenges (CVPR-W)*, 2019 (cit. on p. 60).
- [132] N. Haala, M. Cramer, and K. H. Jacobsen, « The German Camera Evaluation Project - Results from the Geometry Group », in *Canadian Geomatics Conference And Symposium Of Commission I - Geometry*, 2010 (cit. on p. 60).
- [133] F. Rottensteiner, G. Sohn, J. Jung, M. Gerke, C. Baillard, S. Benitez, and U. Breitkopf, « The ISPRS benchmark on urban object classification and 3D building reconstruction », *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 1, pp. 293–298, 2012 (cit. on pp. 60, 63, 144, 152).
- [134] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez, « Can semantic labeling methods generalize to any city? the Inria aerial image labeling benchmark »,

-
- en, in *IEEE International Symposium on Geoscience and Remote Sensing (IGARSS)*, 2017 (cit. on pp. 60, 62, 65).
- [135] I. Demir, K. Koperski, D. Lindenbaum, G. Pang, J. Huang, S. Basu, F. Hughes, D. Tuia, and R. Raskar, « DeepGlobe 2018: a challenge to parse the earth through satellite images », in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPR-W)*, 2018 (cit. on p. 60).
- [136] H. Randrianarivo, B. Le Saux, and M. Ferecatu, « Urban structure detection with deformable part-based models », in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2013 (cit. on pp. 60, 106, 115).
- [137] N. Audebert, B. Le Saux, and S. Lefèvre, « Segment-before-Detect: vehicle detection and classification through semantic segmentation of aerial images », *Remote Sensing*, vol. 9, 4, p. 368, 2017 (cit. on pp. 60, 106, 115).
- [138] R. Daudt, B. Le Saux, A. Boulch, and Y. Gousseau, « Multitask learning for large-scale semantic change detection », *Computer Vision and Image Understanding*, vol. 187, p. 102783, 2019 (cit. on pp. 60, 101).
- [139] M. Schmitt, L. H. Hughes, C. Qiu, and X. X. Zhu, « SEN12MS – a curated dataset of georeferenced multi-spectral sentinel-1/2 imagery for deep learning and data fusion », *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. IV-2/W7, pp. 153–160, 2019 (cit. on p. 60).
- [140] J. Castillo-Navarro, B. Le Saux, A. Boulch, N. Audebert, and S. Lefèvre, « Semi-supervised semantic segmentation in Earth observation: the MiniFrance suite, dataset analysis and multi-task network study », *Machine Learning*, pp. 1–36, 2021 (cit. on pp. 60, 147).
- [141] S. Lobry, B. Demir, and D. Tuia, « RSVQA meets BigEarthNet: a new, large-scale, visual question answering dataset for remote sensing », in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2021 (cit. on p. 60).
- [142] S. Lobry, D. Marcos, J. Murray, and D. Tuia, « RSVQA: visual question answering for remote sensing data », *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, 12, pp. 8555–8566, 2020 (cit. on p. 60).
- [143] A. Torralba and A. A. Efros, « Unbiased look at dataset bias », in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011 (cit. on p. 59).

-
- [144] P. Fisher, A. J. Comber, and R. Wadsworth, « Land use and Land cover: Contradiction or Complement », *Re-presenting GIS*, pp. 85–98, 2005 (cit. on p. 61).
- [145] N. Rey, M. Volpi, S. Joost, and D. Tuia, « Detecting animals in African Savanna with UAVs and the crowds », *Remote Sensing of Environment*, vol. 200, pp. 341–351, 2017 (cit. on p. 62).
- [146] G. Lenczner, A. Chan-Hon-Tong, N. Luminari, B. Le Saux, and G. Le Besnerais, « Interactive learning for semantic segmentation in Earth observation », in *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery Workshops (ECML-PKDD W) - MACLEAN*, 2020 (cit. on p. 62).
- [147] N. Audebert, A. Boulch, H. Randrianarivo, B. Le Saux, M. Ferecatu, S. Lefevre, and R. Marlet, « Deep learning for urban remote sensing », in *Joint Urban Remote Sensing Event (JURSE)*, 2017 (cit. on p. 63).
- [148] N. Audebert, A. Boulch, B. Le Saux, and S. Lefèvre, « Distance transform regression for spatially-aware deep semantic segmentation », *Computer Vision and Image Understanding*, vol. 189, p. 102 809, 2019 (cit. on p. 63).
- [149] L. Mou *et al.*, « Multitemporal Very High Resolution from Space: outcome of the 2016 IEEE GRSS Data Fusion Contest », *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, 8, pp. 3435–3447, 2017 (cit. on p. 65).
- [150] N. Yokoya *et al.*, « Open data for global multimodal land use classification: outcome of the 2017 IEEE GRSS Data Fusion Contest », *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 11, 5, pp. 1363–1377, 2018 (cit. on pp. 65, 144).
- [151] E. Montero, J. Van Wolvelaer, and A. Garzón, « The European Urban Atlas », in *Land Use and Land Cover Mapping in Europe*, 2014, pp. 115–124 (cit. on p. 72).
- [152] A. Lefebvre, C. Sannier, and T. Corpetti, « Monitoring urban areas with Sentinel-2A data: application to the update of the Copernicus high resolution layer imperviousness degree », *Remote Sensing*, vol. 8, 7, p. 606, 2016 (cit. on p. 72).
- [153] L. V. D. Maaten and G. Hinton, « Visualizing Data using t-SNE », *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008 (cit. on p. 77).

-
- [154] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, « Estimating the support of a high-dimensional distribution », *Neural Computation*, vol. 13, 7, pp. 1443–1471, 2001 (cit. on p. 78).
- [155] K. Simonyan and A. Zisserman, « Very deep convolutional networks for large-scale image recognition », in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2015 (cit. on p. 78).
- [156] A. Y. Ng and M. I. Jordan, « On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes », in *Advances in Neural Information Processing Systems (NeurIPS)*, 2002 (cit. on p. 97).
- [157] V. N. Vapnik, *Statistical learning theory*. 1998 (cit. on pp. 97, 121, 133).
- [158] S. Dehaene, *How We Learn: The New Science of Education and the Brain*. 2020 (cit. on pp. 98, 119).
- [159] R. Caruana, « Multitask learning », *Machine learning*, vol. 28, 1, pp. 41–75, 1997 (cit. on pp. 98, 99).
- [160] A. Kumar and H. Daume III, « Learning task grouping and overlap in multi-task learning », in *Proceedings of the International Conference on Machine Learning (ICML)*, 2012 (cit. on p. 99).
- [161] S. Ruder, « An overview of multi-task learning in deep neural networks », *arXiv preprint arXiv:1706.05098*, 2017 (cit. on p. 99).
- [162] S. Ö. Arik, M. Chrzanowski, A. Coates, G. Diamos, A. Gibiansky, Y. Kang, X. Li, J. Miller, A. Ng, J. Raiman, *et al.*, « Deep voice: real-time neural text-to-speech », in *Proceedings of the International Conference on Machine Learning (ICML)*, 2017 (cit. on p. 99).
- [163] B. Romera Paredes, A. Argyriou, N. Bianchi-Berthouze, and M. Pontil, « Exploiting unrelated tasks in multi-task learning », in *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2012 (cit. on p. 99).
- [164] A. R. Zamir, A. Sax, W. Shen, L. J. Guibas, J. Malik, and S. Savarese, « Taskonomy: disentangling task transfer learning », in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018 (cit. on p. 99).

-
- [165] Y. Lu, S. Pirk, J. Dlabal, A. Brohan, A. Pasad, Z. Chen, V. Casser, A. Angelova, and A. Gordon, « Taskology: utilizing task relations at scale », in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021 (cit. on p. 99).
- [166] R. K. Ando, T. Zhang, and P. Bartlett, « A framework for learning predictive structures from multiple tasks and unlabeled data. », *Journal of Machine Learning Research*, vol. 6, 11, 1817-1853, 2005 (cit. on p. 99).
- [167] X. Xia and B. Kulis, « W-Net: a deep model for fully unsupervised image segmentation », *arXiv e-prints*, arXiv:1711.08506, 2017 (cit. on pp. 101, 103, 109).
- [168] W. Chen, Y. Zhang, J. He, Y. Qiao, Y. Chen, H. Shi, and X. Tang, « W-Net: Bridged U-Net for 2D Medical Image Segmentation », *arXiv preprint*, arXiv:1807.04459, 2018 (cit. on p. 101).
- [169] D. Mumford and J. Shah, « Boundary detection by minimizing functionals », in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 1985 (cit. on p. 104).
- [170] B. Kim and J. C. Ye, « Mumford-Shah loss functional for image segmentation with deep learning », *IEEE Transactions on Image Processing*, vol. 29, pp. 1856–1866, 2020 (cit. on pp. 104, 112).
- [171] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, « Context encoders: feature learning by inpainting », in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016 (cit. on p. 105).
- [172] M. Noroozi and P. Favaro, « Unsupervised learning of visual representations by solving jigsaw puzzles », in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016 (cit. on p. 105).
- [173] F. M. Carlucci, A. D’Innocente, S. Bucci, B. Caputo, and T. Tommasi, « Domain generalization by solving jigsaw puzzles », in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019 (cit. on p. 105).
- [174] Z. Chen, V. Badrinarayanan, C.-Y. Lee, and A. Rabinovich, « GradNorm: gradient normalization for adaptive loss balancing in deep multitask networks », in *Proceedings of the International Conference on Machine Learning (ICML)*, 2018 (cit. on p. 117).

-
- [175] P. Bachman, O. Alsharif, and D. Precup, « Learning with pseudo-ensembles », *Advances in Neural Information Processing Systems (NeurIPS)*, 2014 (cit. on p. 120).
- [176] Q. Xie, Z. Dai, E. Hovy, M.-T. Luong, and Q. V. Le, « Unsupervised data augmentation for consistency training », in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020 (cit. on p. 120).
- [177] O. Chapelle, J. Weston, L. Bottou, and V. Vapnik, « Vicinal risk minimization », *Advances in Neural Information Processing Systems (NeurIPS)*, 2001 (cit. on p. 122).
- [178] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, « Mixup: beyond empirical risk minimization », in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018 (cit. on p. 122).
- [179] Y. Grandvalet and Y. Bengio, « Semi-supervised learning by entropy minimization », in *Advances in Neural Information Processing Systems (NeurIPS)*, 2005 (cit. on p. 123).
- [180] E. D. Cubuk, B. Zoph, J. Shlens, and Q. Le, « RandAugment: practical automated data augmentation with a reduced search space », in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020 (cit. on p. 124).
- [181] C. Bishop, *Pattern recognition and machine learning (information science and statistics)*. 2007 (cit. on p. 133).
- [182] P. M. Long, R. A. Servedio, and H. U. Simon, « Discriminative learning can succeed where generative learning fails », *Information Processing Letters*, vol. 103, 4, pp. 131–135, 2007 (cit. on p. 133).
- [183] J. A. Lasserre, C. M. Bishop, and T. P. Minka, « Principled hybrids of generative and discriminative models », in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, 2006 (cit. on p. 133).
- [184] R. Salakhutdinov and G. E. Hinton, « Using deep belief nets to learn covariance kernels for Gaussian processes. », in *Advances in Neural Information and Processing Systems (NeurIPS)*, 2007 (cit. on p. 133).
- [185] S. Bond-Taylor, A. Leach, Y. Long, and C. G. Willcocks, « Deep generative modelling: A comparative review of VAEs, GANs, normalizing flows, energy-based and autoregressive Models », *arXiv preprint arXiv:2103.04922*, 2021 (cit. on p. 134).

-
- [186] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, « Generative adversarial nets », *Advances in Neural Information Processing Systems (NeurIPS)*, 2014 (cit. on p. 134).
- [187] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, « Analyzing and improving the image quality of StyleGAN », in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020 (cit. on p. 134).
- [188] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, « Improved techniques for training GANs », in *Advances in Neural Information Processing Systems (NeurIPS)*, 2016 (cit. on p. 134).
- [189] D. P. Kingma and M. Welling, « An introduction to variational autoencoders », *Foundations and Trends in Machine Learning*, vol. 12, 4, pp. 307–392, 2019 (cit. on p. 135).
- [190] A. Vahdat and J. Kautz, « NVAE: a deep hierarchical variational autoencoder », in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020 (cit. on p. 135).
- [191] A. Nichol and P. Dhariwal, « Improved denoising diffusion probabilistic models », in *Proceedings of the International Conference on Machine Learning (ICML)*, 2021 (cit. on p. 135).
- [192] P. Dhariwal and A. Q. Nichol, « Diffusion models beat GANs on image synthesis », in *Advances in Neural Information Processing Systems (NeurIPS)*, 2021 (cit. on p. 135).
- [193] A. Nichol, P. Dhariwal, A. Ramesh, P. Shyam, P. Mishkin, B. McGrew, I. Sutskever, and M. Chen, « GLIDE: towards photorealistic image generation and editing with text-guided diffusion models », *arXiv preprint arXiv:2112.10741*, 2021 (cit. on p. 135).
- [194] I. Kobyzev, S. Prince, and M. Brubaker, « Normalizing flows: an introduction and review of current methods », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, 11, pp. 3964–3979, 2021 (cit. on p. 135).
- [195] D. P. Kingma and P. Dhariwal, « Glow: generative flow with invertible 1x1 convolutions », in *Advances in Neural Information Processing Systems (NeurIPS)*, 2018 (cit. on p. 136).

-
- [196] A. Pumarola, S. Popov, F. Moreno-Noguer, and V. Ferrari, « C-Flow: conditional generative flow models for images and 3D point clouds », in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020 (cit. on p. 136).
- [197] P. Izmailov, P. Kirichenko, M. Finzi, and A. G. Wilson, « Semi-supervised learning with normalizing flows », in *Proceedings of the International Conference on Machine Learning (ICML)*, 2020 (cit. on p. 136).
- [198] Y. LeCun, S. Chopra, R. Hadsell, M. Ranzato, and F. Huang, « A tutorial on energy-based learning », *Predicting Structured Data*, 2006 (cit. on p. 136).
- [199] Y. Song and D. P. Kingma, « How to train your energy-based models », *arXiv preprint arXiv:2101.03288*, 2021 (cit. on p. 136).
- [200] Y. Du and I. Mordatch, « Implicit generation and modeling with energy based models », in *Advances in Neural Information Processing Systems (NeurIPS)*, 2019 (cit. on pp. 137, 138, 140, 141).
- [201] W. Grathwohl, K.-C. Wang, J.-H. Jacobsen, D. Duvenaud, M. Norouzi, and K. Swersky, « Your classifier is secretly an energy-based model and you should treat it like one », *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020 (cit. on pp. 137, 138, 140, 144).
- [202] S. Geman and D. Geman, « Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images », *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, pp. 721–741, 1984 (cit. on p. 139).
- [203] R. M. Neal *et al.*, « MCMC using Hamiltonian dynamics », *Handbook of Markov Chain Monte Carlo*, vol. 2, 11, p. 2, 2011 (cit. on p. 139).
- [204] M. Welling and Y. W. Teh, « Bayesian learning via stochastic gradient Langevin dynamics », in *Proceedings of the International Conference on Machine Learning (ICML)*, 2011 (cit. on p. 139).
- [205] S. Zhao, J.-H. Jacobsen, and W. Grathwohl, « Joint energy-based models for semi-supervised classification », in *Proceedings of the International Conference on Machine Learning - Workshop on Uncertainty and Robustness in Deep Learning (ICML-W)*, 2020 (cit. on p. 142).

-
- [206] R. Caye Daudt, B. Le Saux, A. Boulch, and Y. Gousseau, « Urban change detection for multispectral Earth observation using convolutional neural networks », in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2018 (cit. on pp. 144, 152).
- [207] S. Zagoruyko and N. Komodakis, « Wide residual networks », in *Proceedings of the British Machine Vision Conference (BMVC)*, 2016 (cit. on p. 144).
- [208] D. P. Kingma and J. Ba, « Adam: a method for stochastic optimization », in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2015 (cit. on p. 144).
- [209] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, « PyTorch: An imperative style, high-performance deep learning library », in *Advances in Neural Information Processing Systems (NeurIPS)*, 2019 (cit. on p. 144).
- [210] D. P. Kingma and M. Welling, « Auto-encoding variational Bayes », in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2014 (cit. on p. 145).
- [211] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, « GANs trained by a two time-scale update rule converge to a local Nash equilibrium », in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017 (cit. on p. 145).
- [212] M. Bińkowski, D. J. Sutherland, M. Arbel, and A. Gretton, « Demystifying MMD GANs », in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018 (cit. on p. 145).
- [213] M. P. Naeini, G. Cooper, and M. Hauskrecht, « Obtaining well calibrated probabilities using bayesian binning », in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2015 (cit. on p. 150).
- [214] D. Hendrycks and K. Gimpel, « A baseline for detecting misclassified and out-of-distribution examples in neural networks », in *Proceedings of International Conference on Learning Representations (ICLR)*, 2017 (cit. on p. 150).
- [215] D. Tuia, C. Persello, and L. Bruzzone, « Domain adaptation for the classification of remote sensing data: an overview of recent advances », *IEEE Geoscience and Remote Sensing Magazine*, vol. 4, 2, pp. 41–57, 2016 (cit. on p. 165).

-
- [216] H. Oliveira, C. Silva, G. L. Machado, K. Nogueira, and J. A. d. Santos, « Fully convolutional open set segmentation », *Machine Learning*, 2021 (cit. on p. [166](#)).

LIST OF FIGURES

1	Semi-supervised learning for large-scale EO data understanding	14
2	Can we get labeled data at a global scale?	21
3	Aguada Fénix, the most ancient Maya temple recently discovered	22
4	Semi-supervised learning for large-scale EO data understanding	26
1.1	Example of different vision tasks	32
1.2	Image segmentation problem	33
1.3	Classical programming vs. machine learning	37
1.4	Supervised vs. semi-supervised learning	39
1.5	Discriminative vs. generative classifiers	40
1.6	Artificial neuron	44
1.7	Single layer perceptron and multilayer perceptron	45
1.8	Standard CNN architecture	47
1.9	Fully convolutional neural network	48
1.10	SegNet architecture	49
1.11	U-Net architecture	50
1.12	Deep semi-supervised learning	50
2.1	Influence of the training set size on the network performances over ISPRS Vaihingen	64
2.2	Semantic maps over Vaihingen, reducing available labeled data for training	65
2.3	Per channel color histograms over the ISPRS Vaihingen dataset	65
2.4	Per channel color histograms on large, multi-location data	67
2.5	Semantic segmentation results on multi-location data	69
2.6	MiniFrance dataset overview	71
2.7	Some samples of MiniFrance dataset on different localizations	73
2.8	Representation of public EO datasets	74
2.9	Subsample for tinyMiniFrance over Cherbourg region	75
2.10	2D representation of images by t-SNE after ResNet34 encoding	77
2.11	Generation of appearance maps for multi-location image datasets	79

2.12	MiniFrance city distributions in the 2D appearance space	80
2.13	IoU and IoT scores between the 2D distributions of MiniFrance cities in the training and the testing splits	81
2.14	Histograms of class distributions by city	83
2.15	MiniFrance class distributions in the 2D appearance space	84
2.16	MiniFrance class distributions aggregated by split	85
2.17	MiniFrance appearance representation aggregated by split	86
2.18	2D representation of images by t-SNE, applied to tinyMiniFrance and Vaihingen together	87
2.19	Some samples of MF-DFC22 dataset on different locations in the training partition	91
3.1	Single task learning and multi-task learning	98
3.2	Proposed multi-task neural network architectures for semi-supervised learning	101
3.3	Classification examples of different methods over tinyMiniFrance	108
3.4	Results comparison for different neural network architectures with re- construction as auxiliary task	111
3.5	Results comparison for different neural networks with unsupervised seg- mentation as auxiliary task	111
3.6	Segmentation maps and reconstruction outputs for BerundaNet-late . .	113
3.7	Semantic segmentation maps and unsupervised segmentation outputs for BerundaNet-late	114
3.8	Semi-supervised results over MiniFrance	116
3.9	Two examples of inference over the CASD dataset	118
3.10	Impact of the λ parameter on the semantic segmentation performance .	119
3.11	Consistency regularization	120
3.12	FixMatch overview	126
4.1	Generative adversarial network	134
4.2	Variational autoencoder	135
4.3	Diffusion models	136
4.4	Normalizing flows	137
4.5	Energy-based models	137
4.6	JEM overview	140

4.7	Generative process by SGLD	142
4.8	Class-conditional samples generated by Joint Energy-based Model (JEM) trained on the EuroSAT dataset	146
4.9	Calibration curves for supervised Wide-ResNet and semi-supervised JEM and FixMatch trained on EuroSAT dataset	150
4.10	JEM log-likelihood (unnormalized) histograms for EuroSAT dataset	151
4.11	Out-of-Distribution detection on different public EO datasets	151
4.12	Semantic maps on never-seen OSCD cities	153
4.13	Semantic maps on never-seen DFC 2017 tile of Rome	154
4.14	Confidence maps obtained by JEM on OSCD and DFC2017 tiles	154

LIST OF TABLES

2.1	Earth Observation datasets summary	60
2.2	Classification performances with respect to amount of data on large, multi-location dataset	66
2.3	Performance by conurbation in the multi-location dataset, training over entire train set	67
2.4	List of cities in MiniFrance and split details	71
2.5	Land use classes available in MiniFrance	73
2.6	Classes distribution on tinyMiniFrance	76
2.7	IoU and IoT scores between training data and test data	86
3.1	Supervised vs. semi-supervised experiments over tinyMiniFrance using different backbone architectures	107
3.2	Neural networks for semi-supervised semantic segmentation comparison	109
3.3	Auxiliary unsupervised loss effect comparison	112
3.4	First semi-supervised results over MiniFrance	115
3.5	Comparison between supervised and semi-supervised methods on CASD	117
3.6	Classification results using FixMatch on two scene classification benchmarks, EuroSAT and So2Sat LCZ42	128
4.1	Models comparison	144
4.2	Classification and generation scores of models trained on EuroSAT . . .	144
4.3	Classification results on EuroSAT. Comparison between Wide-ResNet, BerundaNet, JEM and FixMatch	145
4.4	JEM classification results on different EO datasets	148
4.5	Classification results of a first combination of FixMatch & JEM. Results on EuroSAT	156

Titre : Apprentissage semi-supervisé pour la compréhension des données d'observation de la Terre à large-échelle.

Mot clés : Apprentissage profond, semi-supervision, observation de la Terre, segmentation sémantique, cartographie

Résumé : L'observation de la Terre (OT) joue un rôle important dans la compréhension de notre planète. Aujourd'hui, les données sont facilement accessibles, mais leur volume est tel qu'elles ne peuvent être traitées par des humains. Ainsi, l'intelligence artificielle émerge comme une solution pour le traitement automatique des images d'OT. Cependant, la plupart des données restent sous-exploitées par manque d'annotation sémantique. Par conséquent, l'apprentissage supervisé ne suffit plus pour exploiter pleinement l'information.

Cette thèse étudie des méthodes semi-supervisées (SSL) pour la classification et la segmentation, afin de parvenir à une compréhension des données d'OT à grande échelle. D'abord, nous étudions le potentiel des données non-annotées et proposons des outils pour l'analyse de représentativité pour des bases de données regroupant plusieurs villes. Ensuite, nous explorons deux manières d'abor-

der le SSL : d'un point de vue discriminatif, nous développons des réseaux de neurones multi-tâches et des tâches auxiliaires pour traiter la segmentation sémantique semi-supervisée. Ensuite, nous étudions des méthodes de régularisation par consistance pour effectuer la classification des scènes OT. En ce qui concerne les approches génératives, nous montrons le potentiel d'un modèle conjoint d'énergie (JEM) pour la classification semi-supervisée et pour d'autres applications en OT. Nos expériences montrent que les algorithmes de SSL obtiennent de meilleures performances et offrent des capacités de généralisation pour la cartographie de l'occupation et l'utilisation des sols. Nos contributions portent également sur l'élaboration de MiniFrance, le premier jeu de données ouvert conçu pour évaluer et aider à concevoir des méthodes SSL en télédétection. MiniFrance fait en outre partie de l'IEEE GRSS Data Fusion Contest 2022.

Title: Semi-supervised learning for large-scale Earth observation data understanding.

Keywords: Deep learning, semi-supervised learning, Earth observation, semantic segmentation, land use/land cover mapping.

Abstract: Earth observation (EO) plays a major role in the way we understand our planet and its dynamics. While plenty of data are available, they cannot be processed by humans only, so artificial intelligence has emerged as a solution to achieve automatic analysis of EO imagery. Still, most data are not exploited because they are unlabeled. Hence, algorithms beyond supervised learning are needed to get a complete insight. This thesis investigates deep semi-supervised learning (SSL) for classification and segmentation, in order to achieve EO data understanding at a large-scale. First, we explore the potential of unlabeled data, and propose tools for analyzing data representativeness for multi-location datasets. Then, we explore two ways of approaching the SSL problem. By discriminative modelling,

first we develop multi-task networks and auxiliary tasks to tackle semi-supervised semantic segmentation; second, we explore consistency regularization methods (e.g. FixMatch) to perform scene classification in EO data. Moving to generative modelling, we show the potential of joint energy-based models for semi-supervised classification and many other EO applications. Through extensive experiments, we show that SSL allows us to train algorithms with better performances and generalization capacities for land use and land cover mapping. Finally, our contributions also include the release of MiniFrance, the first dataset and open benchmark designed to assess and help design SSL in remote sensing, and part of the IEEE GRSS Data Fusion Contest 2022.