THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par l'Institut Supérieur de l'Aéronautique et de l'Espace

---

Présentée et soutenue par

Romain THOREAU

Le 20 novembre 2023

Semantic segmentation of airborne hyperspectral images (0.4 - 2.5 µm) for mapping impermeable surfaces in large urban areas

---

École doctorale : **SDU2E - Sciences de l'Univers, de l'Environnement et de l'Espace**

Spécialité : **Surfaces et interfaces continentales, Hydrologie**

Unité de recherche :

ISAE-ONERA PSI Physique Spatiale et Instrumentation

Thèse dirigée par

Xavier BRIOTTET et Laurent RISSER

Rapporteurs :

Mme Céline HUDELOT, CentraleSupélec

M. Paolo GAMBA, Université de Pavie, Italie

Jury :

Mme Céline HUDELOT, CentraleSupélec, Présidente du jury

Mme Yuliya TARABALKA, INRIA / LuxCarta, Examinatrice

M. Karteek ALAHARI, INRIA, Examinateur

M. Patrick GALLINARI, Sorbonne Université / Criteo, Examinateur

M. Laurent RISSER, IMT - CNRS, Co-directeur de thèse

M. Xavier BRIOTTET, ONERA, Directeur de thèse

Mme Véronique ACHARD, ONERA, membre invité

Mme Béatrice BERTHELOT, Magellium, membre invité

Acknowledgment

Pour ces trois années de thèse très épanouissantes, je souhaite tout d'abord remercier chaleureusement mes encadrants. Merci Véronique pour ton implication à toujours discuter du fond avec beaucoup de rigueur et de curiosité intellectuelles. J'admire beaucoup l'écoute et le calme avec lesquels tu considères un problème sous de nouvelles perspectives autant que tu émets des doutes sincères avec une perplexité toujours bienveillante et constructive. Merci pour ta gentillesse et ton attention tout au long de la thèse malgré les circonstances. J'espère que tes qualités scientifiques et humaines auront un peu déteint sur moi, elles continueront de m'inspirer au-delà de la thèse en tout cas ! Merci Laurent d'avoir si promptement rejoint l'encadrement après quelques semaines de thèse. Je crois que ton implication a permis à la thèse d'effectuer un changement sémantique : de « télédétection » à « segmentation » il y a deux communautés scientifiques, deux visions de mêmes problématiques, et je te remercie pour toutes les discussions très structurantes où nous nous sommes demandés comment positionner mes travaux. Merci également pour tes conseils et ton soutien dans la préparation d'une carrière académique ! Merci Béatrice pour ton accompagnement à Magellium et particulièrement dans le pôle physique de la mesure où j'ai appris beaucoup de choses, parfois éloignées de mon sujet de thèse mais toujours passionnantes ! Merci pour la confiance et la liberté que tu m'as accordées. De manière générale, merci à Magellium et à Joël d'avoir co-financé la thèse, j'ai eu beaucoup de plaisir à travailler avec vous ! Merci Xavier pour ton implication dans la thèse, pour la passion avec laquelle tu places la physique au centre de la discussion et pour l'intérêt que tu as porté à la littérature de l'apprentissage automatique. Merci d'avoir eu à la fois un regard « macroscopique » sur la thèse et d'avoir pris le temps d'investiguer très concrètement les données hyperspectrales de Toulouse ! Enfin, un grand merci à tous les quatre pour l'égard que vous avez eu pour mes intérêts scientifiques et professionnels, pour toutes vos relectures, pour vos conseils inestimables et pour m'avoir si souvent fait douter tant « autant que savoir, douter me plaît »[1].

Je souhaite maintenant remercier les rapporteurs et le jury de la thèse. Huge thanks to Céline Hudelot and Paolo Gamba who reviewed my manuscript. I am sincerely grateful for the time you took, for your insightful comments and questions that helped me to improve important aspects of the thesis. Je remercie également tous les membres du jury, Yuliya Tarabalka, Karteek Alahari et Patrick Gallinari. Merci beaucoup pour le regard que vous avez porté sur mes travaux et pour vos questions qui m'ont permis de considérer de nouvelles perspectives ; c'était un honneur et un plaisir !

Beaucoup de personnes m'ont apporté une aide remarquable pendant cette thèse. Je tiens à remercier Nicolas Dobigeon et Nicolas Couellan pour leurs précieux conseils et pour nos interactions lors des comités de suivi ; Chloé Thenoz, Hugo Fournier, Nuria Duran-Gomez, Clément Dechesne, Clément Maliet, Axel Rochel et tous les membres du GT Deep pour les échanges et conseils sur l'ingénierie et la littérature de l'apprentissage automatique ; Thomas Rivière et tous les membres du service informatique de Magellium pour leur support informatique ; Marine François et Isabelle Zanchetta pour leur support administratif ; Gabriel

[1] Dante Alighieri, poète italien du 13e siècle

Calassou et Pierre-Yves Foucher pour avoir pris le temps de discuter des synergies potentielles entre nos travaux ; Laurent Poutier et Philippe Déliot pour avoir traité les images de Toulouse ; Stéphanie Doz pour son aide avec Margit et autres outils ; Philippe Doublet pour le soutien et les discussions sur la thèse; Jean-Philippe Gastellu pour son aide avec le logiciel DART ; Clément Lefranc, Gaspard Loupit et Arthur Cassou dont j'ai eu le plaisir d'encadrer les stages ; Geneviève Soucail pour son soutien infatigable auprès des doctorants de SDU2E ; Etienne Gondet, Nicolas Dobigeon, Emmanuel Rachelson, Adrien Mazoyer, Henrique Goulart, Julie Mauclair, Jérome Mengin, Florence Bannay et Armelle Bonenfant pour m'avoir confié des enseignements et formations pendant la thèse.

Merci à tous les collègues de Magellium et de l'ONERA pour avoir rendu ces trois ans très agréables. Merci à tous les collègues de l'open-space au plus beau sapin de Noël, Florence, Vanessa, Alain, Axel, Antonin, Ornela, Laura, Maryse et Virginie ; à tous les collègues de piscine et de baby-foot, Daria, Lisa, Julien, Nicolas, Kévin, Victor et Victor, Louis, Hélène, Chloé, Théo, Gwenola, Adrien, Nathan, Malo ; aux collègues de Nice, Berlin et La Ciotat, Tristan, Jorge et Guillaume ; et à tous les collègues côtoyés durant ces années ! Merci à tous les permanents, doctorants et stagiaires du DOTA. Merci Mathilda pour ta très grande gentillesse et pour ton énergie incroyable que tu propages tous les jours, merci Laurent pour les conseils de marin aguerri, merci Thierry, Pierre-Yves et Nicolas pour les séances de grimpe. Enfin, merci à tous les « collègues de conférences », en particulier à Iris, Gaston et Arnaud pour les très bons souvenirs à Bonn et à Nice !

En repensant à mes déambulations académiques et professionnelles depuis la fin du lycée, j'éprouve une sincère gratitude pour les corps professoral et administratif qui m'ont accompagné. Parmi toutes les personnes que j'estime pour leur bienveillance et leur engagement auprès des élèves, je souhaiterais remercier en particulier Jean-Marc Rodrigues, François Mendes, Mickaël Prost, Anne Morel, Sandra Cologne, Philippe Viot, Laurent Champaney, Adrien Salomé, Marc Rébillat et Mechbal Nazih.

Je terminerai en remerciant du fond du cœur les personnes qui m'ont entouré ces dernières années. Merci à mes colocataires actuels et passés pour ces douces années à Toulouse, Guillem, Chiara, Sibylle, Anne, Pauline, Gus, Iker, Arthur, Clara, Justine, Lola, Ombeline, Romain et bien sûr tous les amis du Busca ! Merci aux copains du parc des Jards pour toutes les vacances incroyables. Merci à tous les amis du théâtre pour tous les après-midi, soirées et week-ends. Merci aux camarades de promotion, boquets et affiliés, Ilias, Jules, Charlotte, Kiki, Salomé et Louis. J'adore parler de science et de technique avec vous, merci pour toutes ces discussions et surtout merci d'être si présents dans ma vie ! Clo, merci pour les innombrables moments passés ensemble, merci de m'avoir fait parler de mon travail, merci pour tes encouragements, merci pour ces fabuleuses années. Enfin, merci à mes parents, à mon frère et à ma grand-mère pour leur présence et leur inconditionnel soutien tout au long de mes études.

Contents

Abstract

In Earth observation, one of the main challenges is to map soil artificialization, which has been identified as a major public policy issue in the 2018 Biodiversity plan of the French Ministry of Ecological Transition. Airborne hyperspectral imaging has great potential for discriminating the land cover, thanks to its high spatial resolution and high spectral resolution over a wide spectral range. State-of-the-art mapping techniques optimize semantic segmentation models from a subset of labeled pixels, called the training data set. Because labeling pixels require expensive field campaigns and photo-interpretation, the training data set is usually very small with respect to the diversity of materials in metropolitan areas and to the large spectral intra-class variability. In this context, the three main contributions of this thesis consist of leveraging the information of the millions of unlabeled pixels in the hyperspectral image in order to learn discriminating data representations that are robust to intra-class variability (and incidentally to inter-class similarities). Our first contribution is to study the potential of Active Learning (AL) algorithms to improve the quality of the training data set, our second contribution is to integrate a priori physical knowledge in a machine learning model, and our third contribution is to investigate the potential of unsupervised and self-supervised methods for learning discriminating spectral representations.

Key words: hyperspectral imaging, semantic segmentation, machine learning, impermeable surfaces

En observation de la Terre, l'un des principaux défis consiste à cartographier l'artificialisation des sols, qui a été identifiée comme un enjeu majeur de politique publique dans le plan Biodiversité 2018 du ministère de la Transition Écologique. L'imagerie hyperspectrale aéroportée présente un fort potentiel pour discriminer l'occupation du sol, grâce à sa haute résolution spatiale et sa haute résolution spectrale sur un large domaine spectral. Les techniques de cartographie de l'état de l'art optimisent des modèles de segmentation sémantique à partir d'un sous-ensemble de pixels annotés, appelé la base d'apprentissage. Comme l'annotation des pixels nécessite des campagnes terrain et de la photo-interprétation coûteuses, la base d'apprentissage est généralement très petite par rapport à la diversité des matériaux dans les milieux urbains et à la grande variabilité spectrale intra-classe. Dans ce contexte, les trois principales contributions de cette thèse consistent à exploiter l'information des millions de pixels non annotés dans l'image hyperspectrale afin d'apprendre des représentations spectrales discriminantes et robustes aux variabilités intra-classes (et incidemment aux similarités inter-classes). Notre première contribution est d'étudier le potentiel des algorithmes d'Active Learning (AL) pour améliorer la qualité des bases d'apprentissage, notre deuxième contribution est d'intégrer des connaissances physiques a priori dans un modèle d'apprentissage automatique, et notre troisième contribution est d'étudier le potentiel des méthodes non supervisées et auto-supervisées pour l'apprentissage de représentations spectrales discriminantes.

Mots clefs : imagerie hyperspectrale, segmentation sémantique, apprentissage automatique, surfaces imperméables

Short summary in English

In Earth observation, one of the main challenges is to map soil artificialization, which has been identified as a major public policy issue in the 2018 Biodiversity plan of the French Ministry of Ecological Transition. The sealing of natural surfaces by artificial impermeable surfaces has indeed major consequences on watershed hydrology, in particular on floods and droughts, on urban heat island effects and on soil carbon sequestration. If automatic mapping techniques have allowed to produce large scale land cover maps at the object level (e.g. forests, built-up areas, vineyards...) from satellite images, current methods are struggling to produce land cover maps of artificial impermeable surfaces (e.g. concrete, asphalt, paving stones...) at high spatial resolution over large urban areas.

Airborne hyperspectral imaging has great potential for discriminating the land cover, thanks to its high spatial resolution and high spectral resolution over a wide spectral range. State-of-the-art mapping techniques optimize semantic segmentation models (i.e. models that segment the image into zones to which classes are assigned) from a subset of labeled pixels, called the training data set. Because labeling pixels require expensive field campaigns and photo-interpretation, the training data set is usually very small with respect to the diversity of materials in metropolitan areas and to the large spectral intra-class variability, that can be divided into three categories: physics variability induced by different illumination conditions, intrinsic variability induced by slight variations in the material chemical composition and semantic variability resulting from the fact that different materials are actually gathered within the same land cover classes.

In this context, the three main contributions of this thesis consist of leveraging the information of the millions of unlabeled pixels in the hyperspectral image in order to learn discriminating data representations that are robust to intra-class variability (and incidentally to inter-class similarities). Our first contribution is to study the potential of Active Learning (AL) algorithms to select a few additional pixels to annotate in order to improve the quality of the training data set. Our numerical experiments show that different AL strategies are complementary and that we can effectively integrate a priori semantic information through the class hierarchy of impermeable surfaces, resulting in significant improvement of the accuracy of segmentation models for an additional few hundred labeled pixels. Nevertheless, the benefits of AL are limited by the capacity of segmentation models (used by AL methods themselves) to learn discriminating representations. Thus, our second contribution is to introduce a generative model that integrates a priori knowledge derived from physical laws in order to learn, from labeled and unlabeled data, spectral representations that are robust to physics intra-class variations. Our experiments show that our hybrid model has better extrapolation capabilities compared to conventional machine learning models for materials with illumination conditions that were not represented in the training data set (+10% and +2% global accuracy on simulated and real data, respectively). Finally, our third contribution is to investigate the potential of unsupervised and self-supervised methods for learning spectral representations that are robust to intrinsic and semantic variations. To this end, we have built and released a very large hyperspectral database particularly suited to semi-supervised and self-supervised learning, on which we are establishing two baselines for spectral representation learning.

List of Figures

List of Tables

Long résumé en français

1. Contexte

En observation de la Terre, l'un des principaux défis est de cartographier l'artificialisation des sols qui a été identifiée comme un enjeu majeur de politique publique dans le plan Biodiversité 2018 du Ministère de la Transition Écologique [CEV, 2019]. L'étalement urbain, c'est-à-dire l'imperméabilisation de surfaces naturelles avec des matériaux artificiels imperméables, a en effet d'importants impacts environnementaux. Précisément, de nombreuses études scientifiques ont démontré l'influence des surfaces imperméabilisées sur l'hydrologie des bassins versants, et en particulier sur le risque d'inondations et de sécheresses [Bras R.L., 1975; Desbordes, 1989; Dosdogru et al., 2020; Giri et al., 2019; Labbas, 2015], sur les microclimats urbains avec principalement des phénomènes d'îlots de chaleur [Onishi et al., 2010; Pörtner et al., 2022] et sur la captation du carbone par les sols [Commission, 2012; O'Riordan et al., 2021; Pereira et al., 2021; Scalenghe and Marsan, 2009]. Si des techniques de cartographie automatiques ou semi-automatiques ont permis de produire un grand nombre de cartes d'occupation ou d'usage des sols à grandes échelles et à des résolutions spatiales moyennes [Inglada et al., 2017; Stoian et al., 2019; Strand, 2022], aucune ne permet actuellement de cartographier à haute résolution spatiale les différents types de surfaces imperméabilisées à l'échelle d'une métropole. Pourtant, de telles cartes permettraient de guider et prioriser les politiques publiques de désimperméabilisation ou d'atténuation des effets environnementaux de l'imperméabilisation.

Du reste, les instruments de télédétection présentent un grand potentiel pour la cartographie automatique de l'occupation des sols. Les instruments multispectraux embarqués sur des satellites d'observation, tels que Landsat[3] ou Sentinel-2[4], acquièrent des mesures du rayonnement réfléchi par la Terre dans le spectre solaire du visible au moyen infrarouge (sur quelques dizaines de canaux contrairement aux trois canaux classiques RVB), à une résolution spatiale de quelques dizaines de mètres et avec une revisite de quelques jours à une semaine. Les méthodes de vision par ordinateur de l'état de l'art ont démontré de très bonnes capacités à traiter l'information spatiale et spectrale de ces images pour produire de manière automatique 1) des cartes binaires surfaces perméables / surfaces imperméables [Genyun et al., 2015; Sun et al., 2017; Xu, 2008], 2) des cartes de perméabilité (précisément l'Imperviousness Density Map[5] du programme Copernicus[6]), 3) des cartes d'occupation et d'usage des sols dont les nomenclatures comprennent des classes abstraites telles que *Zone densément bâtie*, *Zones industrielles et commerciales*, *Routes* ou encore *Forêt de conifères* [Inglada et al., 2017; Stoian et al., 2019].

A l'opposé, les classes de surfaces imperméabilisées (et par opposition les surfaces perméables) telles que l'*asphalte*, le *gravier* ou le *sol nu* sont peu abstraites. Tandis que l'information spatiale est très peu discriminante pour ces classes, l'information spectrale (intrinsèque à la

[3] https://landsat.gsfc.nasa.gov/satellites/landsat-8/
[4] https://www.esa.int/Applications/Observing_the_Earth/Copernicus/Sentinel-2
[5] https://land.copernicus.eu/pan-european/high-resolution-layers/imperviousness
[6] https://www.copernicus.eu/fr

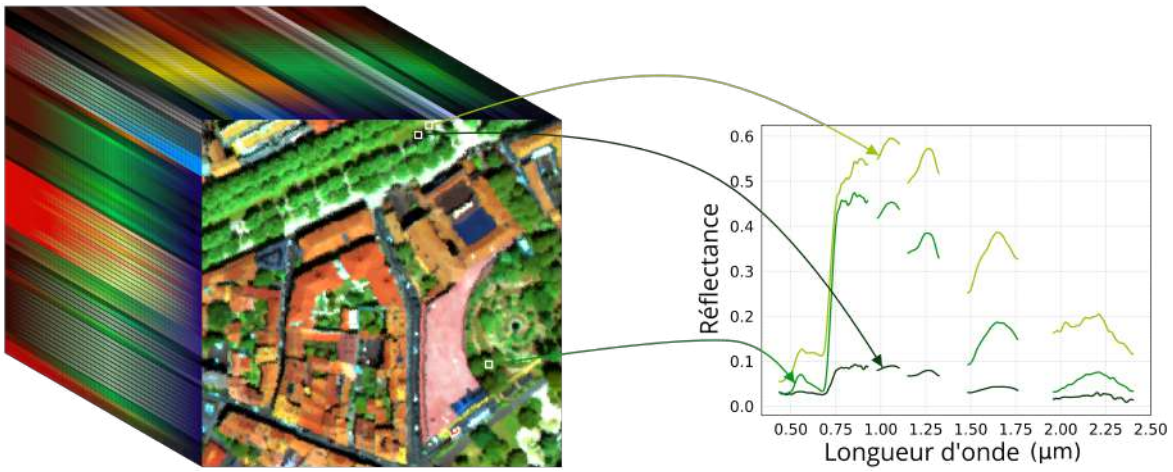


Figure 1: Illustration d'une image hyperspectrale et de différents spectres de végétation

matière) des images multispectrales n'est pas suffisante pour distinguer certains matériaux du fait du faible nombres de canaux et du mélange des signatures spectrales au sein de chaque pixel. Ainsi, cette thèse s'intéresse au traitement d'images hyperspectrales aéroportées dont l'information spectrale offre un très grand potentiel de discrimination de l'occupation des sols. Les images hyperspectrales aéroportées contiennent davantage de canaux que les images multispectrales (quelques centaines de bandes contre quelques dizaines seulement) à de très hautes résolutions spectrales (généralement quelques nanomètres) sur de grand domaines spectraux et à une résolution spatiale de l'ordre du mètre (cf. Fig. 1), ce qui est particulièrement adapté aux milieux urbains qui contiennent une grande diversité de matériaux à de petites échelles (arbres, trottoirs, etc.).

2. Défis

Néanmoins, cartographier de manière automatique les surfaces imperméabilisées d'une métropole à partir d'une image hyperspectrale aéroportée soulève plusieurs défis. Les algorithmes de cartographie de l'état de l'art sont des techniques d'apprentissage automatique qui consistent à optimiser des modèles de segmentation sémantique (*i.e.* un modèle qui segmente l'image en zones auxquelles des classes d'occupation des sols sont attribuées) à partir d'une très faible proportion des pixels de l'image pour lesquels une vérité terrain est connue. Ces pixels annotés avec des classes d'occupation des sols constituent la base d'apprentissage qui est très coûteuse à obtenir à partir de campagnes terrains et de photo-interprétation. Ainsi, étant donnée la faible taille de la base d'apprentissage au regard de la diversité des matériaux en milieu urbain, il est difficile d'optimiser des modèles d'apprentissage statistique qui soient robustes à des variations spectrales intra-classe (et incidemment aux similitudes spectrales inter-classes). La variabilité spectrale intra-classe se distingue en : 1) des variations spectrales dues à des différences d'éclairement (ex : un pixel de végétation à l'ombre ou au soleil), 2) des variations spectrales dues à de faibles changement de la composition chimique d'un matériau (ex : variations de la teneur en eau de la végétation ou vieillissement de l'asphalte) et 3) des variations spectrales dues au fait que différents matériaux sont réunis dans la même classe sémantique (ex : différentes espèces d'arbre réunies dans la même classe *arbre*). On notera dans la suite ces différentes catégories de variabilité intra-classe les variabilités 1) *physiques*, 2) *intrinsèques* et 3) *sémantiques*. Ainsi, le défi est d'induire à partir de la base d'apprentissage des caractéristiques spectrales communes à une classe et discriminantes par rapport aux autres classes. Tandis que différents modèles d'apprentissage automatique pourraient parfaitement s'ajuster à la base d'apprentissage, l'enjeu est d'utiliser une architecture de modèle, une fonction de coût (qui traduit l'erreur de segmentation) et un algorithme

d'optimisation — qui constituent l'ensemble des biais inductifs [Mitchell, 1980; Zhao et al., 2018] — qui permettent de généraliser l'apprentissage à de nouvelles données, c'est-à-dire l'ensemble de l'image.

3. État de l'art

Pour répondre au défi de la variabilité intra-classe, trois axes de recherche indépendants ont suscité beaucoup d'attention dans la littérature : 1) l'architecture des modèles de segmentation, 2) la qualité des bases d'apprentissage et 3) les algorithmes d'optimisation.

3.1. Architecture des modèles de segmentation

La segmentation sémantique d'images hyperspectrales se distingue en deux approches : la classification de pixels à partir de l'information spectrale uniquement, par exemple avec des Forêts Aléatoires [Breiman, 2001] ou des Machines à Vecteurs Supports [Cortes and Vapnik, 1995] qui ont été très étudiées dans la communauté, et la classification simultanée des pixels d'une imagette (typiquement de 16×16 pixels). Récemment, les réseaux de neurones à convolutions spatiales-spectrales (CNNs 3D) sont devenus l'état de l'art [Audebert et al., 2019; Li et al., 2017; Rasti et al., 2020]. Les CNNs 3D apprennent simultanément à représenter les données de manière fortement non linéaire dans un espace de faible dimension, depuis lequel des frontières de décision linéaires séparent les données selon leurs classes sémantiques. Ces modèles ont obtenu de très hautes précisions de segmentation sur des données hyperspectrales de référence comme Pavia University[7], Indian Pines[12] et Botswana[12] qui sont couramment utilisées dans la littérature pour comparer les modèles de segmentation sémantique. Néanmoins, les capacités de généralisation des CNNs 3D n'ont pas encore été démontrées sur des images plus grandes, avec davantage de classes et de variabilité spectrale intra-classe.

3.2. Qualité de la base d'apprentissage

L'objectif des méthodes d'Active Learning est d'améliorer la base d'apprentissage en annotant quelques pixels supplémentaires de manière itérative par l'interaction entre un algorithme qui sélectionne les pixels les plus informatifs pour la tâche de segmentation et un utilisateur qui annote ces pixels [Settles, 2012]. Différentes stratégies d'Active Learning ont été développées pour la segmentation sémantique d'images hyperspectrales, dont [Di and Crawford, 2010; Li et al., 2011; Rajan et al., 2008], qui ont empiriquement montré sur des données de référence que quelques centaines de pixels additionnels pouvaient significativement améliorer les performances de segmentation. Néanmoins, l'utilisation de méthodes d'Active Learning dans un contexte opérationnel pose plusieurs difficultés qui n'ont pas été adressées : le nombre de pixels non annotés est considérablement plus grand que sur les images de référence couramment utilisées, la nomenclature initiale n'est souvent pas assez exhaustive pour représenter la diversité des matériaux présents sur l'image et la variabilité spectrale intra-classe est généralement plus grande.

3.3. Algorithmes d'optimisation

Afin d'améliorer l'optimisation des modèles de segmentation à partir de peu de données annotées, des techniques d'optimisation qui ne requièrent pas d'annotations telles que [Hendrycks et al., 2019; Laine and Aila, 2016] ont été adaptées à l'imagerie hyperspectrale [Liu et al., 2017; Rasti et al., 2020; Yue et al., 2021]. Les résultats expérimentaux sur de petites images hyperspectrales ont montré que les méthodes non-supervisées ou semi-supervisées pouvaient surpasser les algorithmes supervisés en terme de précision de segmentation. Leur potentiel

[7] https://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes

n'a cependant pas été étudié sur de grandes images hyperspectrales où de nombreux pixels non annotés n'appartiennent à aucune classe de la base d'apprentissage.

4. Contributions

Étant données les limitations actuelles dans la littérature, l'objectif de cette thèse est de développer une méthodologie pour automatiquement cartographier les surfaces imperméabilisées d'une métropole à partir d'une image hyperspectrale aéroportée. Précisément, nous souhaitons répondre au problème du peu de disponibilité de données annotées en tirant profit de l'information des dizaines de millions de pixels non annotés d'une image hyperspectrale selon trois approches, qui constituent nos contributions.

4.1. Chapitre 4

Notre première contribution est d'étudier le potentiel des méthodes d'Active Learning (AL) à sélectionner parmi tous les pixels de l'image quelques centaines à annoter pour améliorer 1) la nomenclature des cartes des surfaces imperméabilisées et 2) la *qualité* des bases d'apprentissage.

Synthèse des stratégies d'AL et unification du formalisme d'AL Nous proposons de classer les méthodes d'AL en trois catégories : les heuristiques d'incertitudes, les heuristiques de représentativité et les heuristiques de performance. Les heuristiques d'incertitudes considèrent que les pixels les plus informatifs sont ceux pour lesquels le modèle de segmentation est le plus incertain, tandis que les heuristiques de représentativité considèrent que les pixels les plus informatifs sont ceux qui représentent au mieux l'ensemble des pixels dans l'image. Finalement, les heuristiques de performance considèrent que les pixels les plus informatifs sont ceux qui permettent la plus grande réduction de l'erreur de généralisation. Ainsi, nous avons sélectionné sept méthodes de la littérature (selon l'intérêt qu'elles ont suscité et selon leur ancienneté) dont nous avons unifié le formalisme : Breaking Ties [Tong Luo et al., 2004], BALD [Houlsby et al., 2011], Batch-BALD [Kirsch et al., 2019] (heuristiques d'incertitudes), Core-Set [Sener and Savarese, 2018], Hierarchical Sampling [Dasgupta and Hsu, 2008], VAAL [Sinha et al., 2019] (heuristiques de représentativité) et LAL [Konyushkova et al., 2017] (heuristique de performance).

Étude empirique du potentiel de l'AL Nous avons quantifié la capacité des méthodes d'AL retenues à significativement améliorer la cartographie des surfaces imperméabilisées d'une métropole en mesurant deux métriques : la précision de segmentation après l'ajout de pixels annotés à chaque itération de l'algorithme et le nombre de pixels additionnels par classe (y compris parmi les nouvelles classes découvertes au cours des itérations). Nos expériences sur trois images hyperspectrales ont montré que différentes stratégies d'Active Learning sont complémentaires et peuvent considérablement aider ingénieurs et chercheurs pendant des campagnes terrains pour 1) découvrir des matériaux peu représentés sur l'image mais ayant une importance pour la cartographie avec la méthode BALD [Houlsby et al., 2011], 2) sélectionner des pixels très informatifs des différences entre des classes spectralement similaires avec la méthode Breaking Ties [Tong Luo et al., 2004] et 3) améliorer la représentativité de la variabilité intra-classe en très peu d'itérations avec la méthode Core-Set [Sener and Savarese, 2018]. Dans un cas opérationnel, des gains de précision d'au moins 10% par rapport à un tirage aléatoire ont été obtenus avec ces méthodes en annotant 300 pixels supplémentaires seulement.

Réduction des besoins computationnels des techniques d'AL par pré-traitement des données Afin de réduire le temps de calcul et le besoin en mémoire de certaines méthodes qui entravent leur utilisation lors de campagnes terrains, nous avons développé plusieurs

méthodes de pré-traitement des images hyperspectrales et évalué leur influence sur les performances de l'Active Learning. Les méthodes de pré-traitement introduites segmentent l'image en super-pixels, sur lesquels les méthodes d'AL sont appliquées. Des pixels sont alors aléatoirement échantillonnés parmi les super-pixels les plus informatifs. Nos expériences montrent que le prétraitement réduit significativement le temps de calcul des méthodes (jusqu'à 25 fois plus rapide pour la méthode Core-set) sans dégrader les performances de l'algorithme en terme de précision de segmentation.

Intégration d'a priori sémantique Finalement, une limitation des méthodes d'AL est qu'elles considèrent chaque classe comme ayant la même importance. Cependant, certaines confusions entre différentes surfaces imperméabilisées sont plus importantes que d'autres : plus la différence de perméabilité entre deux classes est grande, plus la confusion importe. Nous avons donc développé la méthode Probabilistic Breaking Ties, une amélioration de la méthode Breaking Ties qui intègre cet a priori sémantique pour réduire les confusions entre des matériaux avec de grandes différences de perméabilité. Dans nos expériences, Probabilistic Breaking Ties permet de réduire par deux le nombre d'itérations nécessaires pour atteindre les même performances que Breaking Ties en terme de précision de classification des surfaces perméables vs imperméables.

4.2. Chapitre 5

Si les algorithmes d'Active Learning peuvent exploiter des heuristiques statistiques pour guider le choix de quelques pixels additionnels à annoter, leur potentiel est limité par la capacité des modèles de segmentation (utilisés par les méthodes d'AL elles-mêmes) à apprendre des représentations discriminantes. Ainsi, notre seconde contribution est d'utiliser à la fois les données non annotées et des biais déductifs dérivés de lois physiques pour apprendre des représentations spectrales robustes à des variations intra-classe *physiques*.

Intégration d'un modèle physique dans un VAE : un formalisme général Nous avons développé p^{3}VAE, un auto-encodeur variationnel qui intègre un modèle physique dans le décodeur. L'objectif de p^{3}VAE est de découpler les facteurs de variations qui sont intrinsèques aux matériaux des facteurs de variations liés aux conditions d'acquisition. La combinaison de modèles physiques et de modèles statistiques est récemment devenue un sujet de recherche très actif, dont les avantages en terme de capacité d'extrapolation et d'interprétabilité ont été démontrés par de premiers travaux [Raissi et al., 2019; Takeishi and Kalousis, 2021]. En particulier, [Aragon-Calvo and Carvajal, 2020] sont les premiers, à notre connaissance, à avoir substitué un modèle physique à la place du décodeur d'un auto-encodeur pour inverser un modèle physique analytique. [Takeishi and Kalousis, 2021] ont généralisé les travaux de [Aragon-Calvo and Carvajal, 2020] en développant un formalisme appelé *physics-integrated VAE* qui complémente un modèle physique imparfait avec un réseau de neurones dans le décodeur d'un VAE. Une technique de régularisation, essentielle à leur contribution, permet de limiter la capacité de représentation du réseau de neurones pour conserver une utilisation cohérente du modèle physique. Au contraire, p^{3}VAE intègre un modèle **p**hysique supposé **p**arfait qui explique **p**artiellement les facteurs de variation des données. En d'autres termes, nous considérons les cas où le modèle direct est partiellement connu : le modèle physique, à lui seul, ne peut approximer la distribution des données, mais peut avec précision exprimer la dépendance d'un sous-ensemble des facteurs de variations aux observations.

Application de p^{3}VAE à la segmentation sémantique d'images hyperspectrales Nous avons défini un modèle de transfert radiatif simplifié qui exprime les variations spectrales physiques comme une fonction du rapport des conditions d'éclairement réelles (qui dépendent de la topographie) sur les conditions d'éclairement supposées (c'est à dire l'éclairement

que l'on aurait pour un sol plat). Le décodeur de p^{3}VAE est une combinaison de ce modèle physique et d'un réseau de neurones. En fait, la partie physique de décodeur peut être considérée comme une couche dense classique dont les poids et le biais sont fixés par le modèle physique. L'espace latent de p^{3}VAE est implicitement divisé en deux sous-ensemble : un sous-ensemble qui correspond aux variables physiques, c'est à dire les conditions d'éclairement, et un sous-ensemble qui n'a pas de sens physique et correspond ici aux classes ainsi qu'aux variabilités intra-classe intrinsèque et sémantique. Ce désenchevêtrement de l'espace latent est favorisé par le modèle physique qui prend en entrée le premier sous-ensemble comme s'il correspondait réellement aux conditions d'éclairement. Afin d'évaluer p^{3}VAE en termes de précision, d'interprétabilité et de désenchevêtrement, nous avons simulé une base de données hyperspectrales avec différentes conditions d'éclairement avec le logiciel DART [Gastellu-Etchegorry et al., 2012]. Pour valider le modèle physique simplifié dans des conditions réelles, nous avons également constitué une vérité terrain sur une image hyperspectrale. Nous avons comparé les performances de p^{3}VAE à des modèles génératifs semi-supervisés classiques, incluant un VAE [Kingma et al., 2014] et un GAN [Spurr et al., 2017] semi-supervisés, ainsi que des modèles de segmentation classiques supervisés dont FG-Unet [Stoian et al., 2019], un réseau de neurones à convolutions totalement connectées. Les expériences numériques montrent que p^{3}VAE a une capacité d'extrapolation supérieure aux modèles d'apprentissage conventionnels pour des matériaux avec des conditions d'éclairement qui n'étaient pas représentées dans la base d'apprentissage (+10% et +2% de précision en moyenne sur les données simulées et réelles, respectivement).

4.3. Chapitre 6

Dans un cas opérationnel, le nombre de pixels annotés est si faible au regard du nombre de pixels sur l'image que les méthodes d'Active Learning, même combinées avec le modèle hybride p^{3}VAE, ne pourraient aboutir à une base d'apprentissage totalement représentative des variabilités intra-classe *intrinsèque* et *sémantique*. Notre troisième contribution est donc d'étudier le potentiel des méthodes non-supervisées et auto-supervisées pour apprendre des représentations spectrales robustes aux variations *intrinsèque* et *sémantique*.

Introduction et analyse d'une grande base de données hyperspectrales Afin de pouvoir correctement comparer et étudier les approches non-supervisées et auto-supervisées, nous avons introduit une nouvelle base de données hyperspectrales à partir d'une image aéroportée de Toulouse, France, qui se démarque des jeux de données actuellement utilisés dans la littérature par les aspects suivants :

- Une très haute résolution spectrale ($\leq$ 12 nm) de 0.4 µm à 2.5 µm et une très haute résolution spatiale (1 m),
- Une grande nomenclature hiérarchique contenant 32 classes d'occupation des sols, divisés en 16 matériaux perméables et 16 matériaux imperméables,
- 8 ensembles d'apprentissage (comprenant un ensemble pour une partie supervisée et un ensemble pour une partie non supervisée), de validation et de test spatialement disjoints,
- Environ 380,000 annotations éparses sur une zone de 90 km^2.

Une étude qualitative montre que le jeu de Toulouse est plus complexe et plus représentatif de la diversité spectrale d'un milieu urbain que les autres jeux de données disponibles dans la littérature. En outre, nous pensons que les ensembles d'apprentissage, de validation et de test fournis avec les données permettront une comparaison équitable des méthodes de la littérature tandis que les autres images hyperspectrales couramment utilisées n'ont pas d'ensembles de référence pour l'évaluation de techniques semi-supervisées, ce qui conduit chaque auteur

à proposer le sien et nuit à la comparaison équitable des méthodes.

Une étude préliminaire sur les méthodes non-supervisées et auto-supervisées Les méthodes auto-supervisées consistent à pré-entraîner des modèles de segmentation sémantique sur d'autres tâches que de la segmentation sémantique. Ces tâches prétextes, pour lesquelles une vérité terrain peut être automatiquement générée, permettent aux modèles d'apprendre à représenter les données dans un espace de faible dimension, sur lequel les modèles pourront ensuite être optimisés plus finement pour la segmentation. Tandis que de nombreuses tâches auto-supervisées sont fondées sur des techniques d'augmentation de données, qui en substance génèrent de nouveaux exemples d'une même donnée avec des contextes différents (par exemple un chat sur une image avec une orientation différente et un arrière plan différent), il semble difficile d'augmenter des données hyperspectrales tels que les exemples générés soient conformes à des variations spectrales *intrinsèque* et *sémantique*. C'est pour cette raison, et avec l'idée que certaines combinaisons de caractéristiques spectrales (par exemple un pic d'absorption dans l'infrarouge et une grande pente dans le visible) sont très informatives du matériau, que nous avons adapté la technique auto-supervisée [He et al., 2022] qui consiste à encoder dans un espace latent des données partiellement masquées et à reconstruire les parties manquantes. Des expériences comparatives sur les données de Toulouse montrent que l'apprentissage de représentation non-supervisée avec un auto-encodeur classique ou l'apprentissage de représentation auto-supervisée avec l'adaptation de [He et al., 2022] combinés avec une forêt aléatoire [Breiman, 2001] surpassent significativement les performances de plusieurs modèles de référence (de +5% à 20% de gain de précision).

Ensemble de réseaux prototypaux pour une approche semi-supervisée Les réseaux prototypaux, initialement introduits pour des tâches de *few-shot classification*, ont des propriétés intéressantes pour partiellement superviser l'apprentissage de représentation. Nous montrons que des ensembles de réseaux prototypaux peuvent être facilement créés pour des tâches classiques de classification. Néanmoins, les expériences numériques sur la combinaison des méthodes auto-supervisées avec des réseaux prototypaux ne sont pas encore abouties.

5. Publications

Les travaux de cette thèse ont été concrétisés par des publications dans plusieurs journaux internationaux à comité de lecture ainsi que plusieurs communications au cours de conférences internationales.

Articles publiés ou en cours de révision dans des journaux internationaux à comité de lecture

- R. Thoreau, V. Achard, L. Risser, B. Berthelot and X. Briottet, "Active Learning for Hyperspectral Image Classification: A comparative review," in IEEE Geoscience and Remote Sensing Magazine, vol. 10, no. 3, pp. 256-278, Sept. 2022, doi: 10.1109/MGRS.2022.3169947,
- R. Thoreau, L. Risser, V. Achard, B. Berthelot and X. Briottet, "p^{3}VAE: a physics-integrated generative model. Application to the semantic segmentation of optical remote sensing images," arXiv preprint arxiv.org/abs/2210.10418.
- R. Thoreau, L. Risser, V. Achard, B. Berthelot and X. Briottet, "Toulouse Hyperspectral Data Set: a benchmark data set to assess semi-supervised spectral representation learning and pixel-wise classification techniques," arXiv preprint arxiv.org/abs/2311.08863.

Actes de conférences internationales à comité de lecture

- R. Thoreau, V. Achard, L. Risser, B. Berthelot and X. Briottet, "Active Learning for Hyperspectral Image Classification: A comparative review," in IEEE Geoscience and Remote Sensing Magazine, vol. 10, no. 3, pp. 256-278, Sept. 2022, doi: 10.1109/MGRS.2022.3169947,
- R. Thoreau, V. Achard, L. Risser, B. Berthelot and X. Briottet, "Active Learning On Large Hyperspectral Datasets: A preprocessing method", Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci., XLIII-B3-2022, 435–442, doi: 10.5194/isprs-archives-XLIII-B3-2022-435-2022,2022,
- R. Thoreau, V. Achard, L. Risser, B. Berthelot and X. Briottet, "Probabilistic Breaking Tie: An Active Learning Strategy To Leverage Class Hierarchy For Impervious Surfaces Classification," 2022 12th Workshop on Hyperspectral Imaging and Signal Processing: Evolution in Remote Sensing (WHISPERS), Rome, Italy, 2022, pp. 1-5, doi: 10.1109/WHISPERS56178.2022.9955057.

References

Aragon-Calvo, M. A. and Carvajal, J. (2020). Self-supervised learning with physics-aware neural networks–i. galaxy model fitting. *Monthly Notices of the Royal Astronomical Society*, 498(3):3713–3719. 5

Audebert, N., Le Saux, B., and Lefèvre, S. (2019). Deep learning for classification of hyperspectral data: A comparative review. *IEEE geoscience and remote sensing magazine*, 7(2):159–173. 3

Bras R.L., P. F. (1975). Effects of urbanization on catchment response. *Journal of Hydraulics Division*, 101:451–466. 1

Breiman, L. (2001). Random forests. *Machine learning*, 45:5–32. 3, 7

CEV, C. p. l. V. (2019). Les enjeux de l'artificialisation des sols : diagnostic. https://www.ecologie.gouv.fr/sites/default/files/Les enjeux de l'artificialisation des sols.pdf. 1

Commission, E. E. (2012). Guidelines on best practice to limit, mitigate or compensate soil sealing. *Luxembourg: European Union SWD (2012) 101*. 1

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20:273–297. 3

Dasgupta, S. and Hsu, D. (2008). Hierarchical sampling for active learning. In *Proceedings of the 25th international conference on Machine learning*, pages 208–215. 4

Desbordes, M. (1989). Principales causes d'aggravation des dommages dus aux inondations par ruissellement superficiel en milieu urbanisé. *Bulletin hydrologie urbaine*, 4:2–10. 1

Di, W. and Crawford, M. M. (2010). Multi-view adaptive disagreement based active learning for hyperspectral image classification. In *2010 IEEE International Geoscience and Remote Sensing Symposium*, pages 1374–1377. 3

Dosdogru, F., Kalin, L., Wang, R., and Yen, H. (2020). Potential impacts of land use/cover and climate changes on ecologically relevant flows. *Journal of Hydrology*, 584:124654. 1

Gastellu-Etchegorry, J.-P., Grau, E., and Lauret, N. (2012). Dart: A 3d model for remote sensing images and radiative budget of earth surfaces. *Modeling and simulation in engineering*, (2). 6

Genyun, S., Chen, X., Jia, X., Yao, Y., and Wang, Z. (2015). Combinational build-up index (cbi) for effective impervious surface mapping in urban areas. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 9:1–12. 1

Giri, S., Zhang, Z., Krasnuk, D., and Lathrop, R. G. (2019). Evaluating the impact of land uses on stream integrity using machine learning algorithms. *Science of The Total Environment*, 696:133858. 1

He, K., Chen, X., Xie, S., Li, Y., Dollár, P., and Girshick, R. (2022). Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009. 7

Hendrycks, D., Mazeika, M., Kadavath, S., and Song, D. (2019). Using self-supervised learning can improve model robustness and uncertainty. *Advances in neural information processing systems*, 32. 3

Houlsby, N., Huszár, F., Ghahramani, Z., and Lengyel, M. (2011). Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*. 4

Inglada, J., Vincent, A., Arias, M., Tardy, B., Morin, D., and Rodes, I. (2017). Operational high resolution land cover map production at the country scale using satellite image time series. *Remote Sensing*, 9(1):95. 1

Kingma, D. P., Mohamed, S., Jimenez Rezende, D., and Welling, M. (2014). Semi-supervised learning with deep generative models. *Advances in neural information processing systems*, 27. 6

Kirsch, A., Van Amersfoort, J., and Gal, Y. (2019). Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning. *Advances in neural information processing systems*, 32. 4

Konyushkova, K., Sznitman, R., and Fua, P. (2017). Learning active learning from data. *Advances in neural information processing systems*, 30. 4

Labbas, M. (2015). *Modélisation hydrologique de bassins versants périurbains et influence de l'occupation du sol et de la gestion des eaux pluviales. Application au bassin de l'Yzeron (130 km2)*. Theses, Doctorat, spécialité : Océan, Atmosphère, Hydrologie, Université de Grenoble. 1

Laine, S. and Aila, T. (2016). Temporal ensembling for semi-supervised learning. In *International Conference on Learning Representations*. 3

Li, J., Bioucas-Dias, J. M., and Plaza, A. (2011). Hyperspectral image segmentation using a new bayesian approach with active learning. *IEEE Transactions on Geoscience and Remote Sensing*, 49(10):3947–3960. 3

Li, Y., Zhang, H., and Shen, Q. (2017). Spectral–spatial classification of hyperspectral imagery with 3d convolutional neural network. *Remote Sensing*, 9(1):67. 3

Liu, B., Yu, X., Zhang, P., Tan, X., Yu, A., and Xue, Z. (2017). A semi-supervised convolutional neural network for hyperspectral image classification. *Remote Sensing Letters*, 8(9):839–848. 3

Mitchell, T. M. (1980). *The need for biases in learning generalizations*. 3

Onishi, A., Cao, X., Ito, T., Shi, F., and Imura, H. (2010). Evaluating the potential for urban heat-island mitigation by greening parking lots. *Urban Forestry and Urban Greening*, 9(4):323 – 332. 1

O'Riordan, R., Davies, J., Stevens, C., and Quinton, J. N. (2021). The effects of sealing on urban soil carbon and nutrients. *SOIL*, 7(2):661–675. 1

Pereira, M. C., O'Riordan, R., and Stevens, C. (2021). Urban soil microbial community and microbial-related carbon storage are severely limited by sealing. *Journal of Soils and Sediments*, 21:1455–1465. 1

Pörtner, H.-., Roberts, D., Tignor, M., E.S., P., Mintenbeck, K., Alegría, A., Craig, S., Langsdorf, S., Löschke, S., Möller, V., Okem, A., Rama, B., and eds (2022). *Climate Change 2022: Impacts, Adaptation and Vulnerability. Contribution of Working Group II to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change*, volume 1. Cambridge University Press. 1

Raissi, M., Perdikaris, P., and Karniadakis, G. E. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378:686–707. 5

Rajan, S., Ghosh, J., and Crawford, M. M. (2008). An active learning approach to hyperspectral data classification. *IEEE Transactions on Geoscience and Remote Sensing*, 46(4):1231–1242. 3

Rasti, B., Hong, D., Hang, R., Ghamisi, P., Kang, X., Chanussot, J., and Benediktsson, J. A. (2020). Feature extraction for hyperspectral imagery: The evolution from shallow to deep: Overview and toolbox. *IEEE Geoscience and Remote Sensing Magazine*, 8(4):60–88. 3

Scalenghe, R. and Marsan, F. A. (2009). The anthropogenic sealing of soils in urban areas. *Landscape and urban planning*, 90(1-2):1–10. 1

Sener, O. and Savarese, S. (2018). Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*. 4

Settles, B. (2012). *Active Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan and Claypool Publishers. 3

Sinha, S., Ebrahimi, S., and Darrell, T. (2019). Variational adversarial active learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5972–5981. 4

Spurr, A., Aksan, E., and Hilliges, O. (2017). Guiding infogan with semi-supervision. *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 119–134. 6

Stoian, A., Poulain, V., Inglada, J., Poughon, V., and Derksen, D. (2019). Land cover maps production with high resolution satellite image time series and convolutional neural networks: Adaptations and limits for operational systems. *Remote Sensing*, 11(17):1986. 1, 6

Strand, G.-H. (2022). Accuracy of the copernicus high-resolution layer imperviousness density (hrl imd) assessed by point sampling within pixels. *Remote Sensing*, 14(15):3589. 1

Sun, Z., Wang, C., Guo, H., and Shang, R. (2017). A modified normalized difference impervious surface index (mndisi) for automatic urban mapping from landsat imagery. *Remote Sensing*, 9:942. 1

Takeishi, N. and Kalousis, A. (2021). Physics-integrated variational autoencoders for robust and interpretable generative modeling. *Advances in Neural Information Processing Systems*, 34:14809–14821. 5

Tong Luo, Kramer, K., Samson, S., Remsen, A., Goldgof, D. B., Hall, L. O., and Hopkins, T. (2004). Active learning to recognize multiple types of plankton. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, volume 3, pages 478–481 Vol.3. 4

Xu, H. (2008). A new index for delineating built-up land features in satellite imagery. *International Journal of Remote Sensing*, 29:4269 – 4276. 1

Yue, J., Fang, L., Rahmani, H., and Ghamisi, P. (2021). Self-supervised learning with adaptive distillation for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–13. 3

Zhao, S., Ren, H., Yuan, A., Song, J., Goodman, N., and Ermon, S. (2018). Bias and generalization in deep generative models: An empirical study. *Advances in Neural Information Processing Systems*, 31. 3

Chapter 1

Introduction

Contents

In Earth observation, one of the main challenges is to map soil artificialization, which was identified as a major public policy issue in the 2018 Biodiversity plan [CEV, 2019] of the French Ministry of Ecological Transition. Urban sprawl, which consists in the sealing of natural surfaces with artificial impermeable surfaces, indeed has major environmental consequences. Scientific studies have shown the impact of artificial impermeable surfaces on watershed hydrology, urban micro-climates and carbon cycle [Dosdogru et al., 2020; Giri et al., 2019; Labbas, 2015; Pörtner et al., 2022]. If semi-automatic mapping techniques have allowed to produce large scale permeability maps at medium spatial resolution [Strand, 2022], current algorithms are unable to produce exhaustive land cover maps of artificial impermeable surfaces at high spatial resolution over metropolises. Therefore, public authorities cannot design relevant policies to mitigate the environmental consequences of artificial impermeable surfaces.

To fully understand the motivation of this thesis, we define what are impermeable surfaces and present the impact of surface sealing on the environment in section 1. We describe the current available tools to monitor urban sprawl and the scientific barriers that prevent the detailed mapping of artificial impermeable surfaces over metropolises in section 2, which finally brings us to the objectives of this thesis in section 3. We present the organization of the manuscript in section 4.

1 Impermeable surfaces and impact of surface sealing on the environment

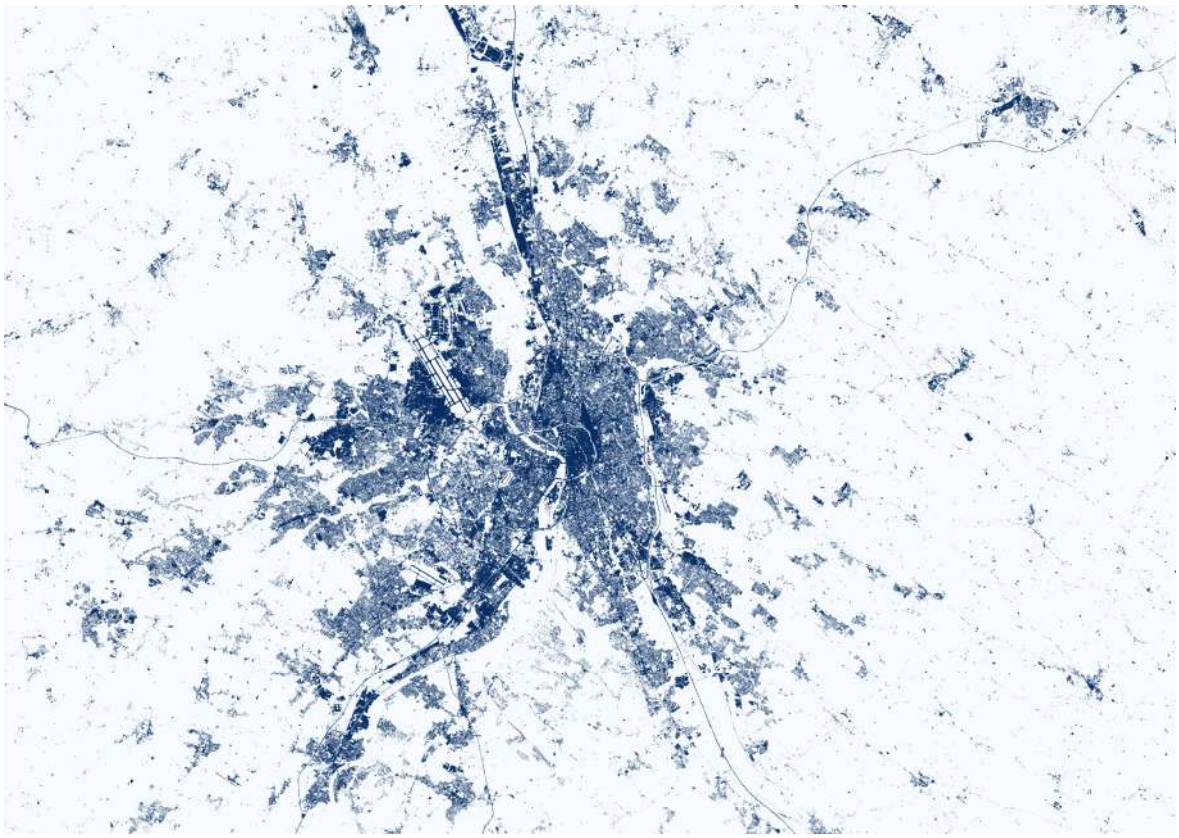


Figure 1.1: Urban Atlas Imperviousness Density Map[1]over Toulouse, France

1.1 Impermeable surfaces

Permeability is a property of materials that indicates how well a fluid can flow through them. More precisely, the flow velocity of a fluid in the soil depends on the soil permeability, the fluid properties (dynamic viscosity and bulk density) and the pressure drop [Magnan, 1999]. In 1856, Henry Darcy experimentally determined the relationship between the water flow velocity and the hydraulic load within a material by introducing a permeability coefficient, homogeneous to a velocity. For instance, the permeability of porous gravels is 1 m/s while the permeability of clay is 10^{-11} m/s [Magnan, 1999]. In the literature, impermeable surfaces refer to surfaces with a low permeability coefficient[2]. In particular, artificial surfaces such as asphalt, bitumen, tar or tile are impermeable. Simpler definitions can be found: the Côte Basque - Adour territorial collectivity defines permeable surfaces as "geographic spaces exclusively occupied by chlorophyllous vegetation that allows the infiltration of rainwater" and impermeable surfaces as "geographic areas that are not covered by vegetation".
On the contrary, more complex distinctions exist in hydrology where impermeable surfaces may be divided into two categories: total impermeable surfaces and effective impermeable surfaces [Labbas, 2015]. Effective impermeable surfaces, in contrast to total impermeable surfaces, are connected to a drainage system. For instance, roofs can be directly connected to the sewers. While it is beyond the scope of this thesis to distinguish these surfaces, it

[1] https://land.copernicus.eu/pan-european/high-resolution-layers/imperviousness/
[2] However, there is no consensus on the threshold at which surfaces are impermeable.

should be kept in mind that a more comprehensive soil material mapping than permeable - impermeable classification could be useful.

Between 1982 and 2012, the portion of artificial impermeable surfaces in metropolitan France increased by 62%, with a significant acceleration from the 2000s [CEV, 2019]. Worldwide, [Sun et al., 2022] estimated from satellite images a 1.6% growth of global artificial impermeable surfaces from 2015 to 2018 and observed significant differences between continents. For instance, impermeable surfaces in South America increased by 3.3% over the same period while China, USA and Russia approximately accounted for 50% of the global artificial impermeable surfaces in 2018 [Sun et al., 2022].

1.2 Impact of surface sealing on watershed hydrology

Impermeable surfaces affect the water balance and response times of watersheds by altering water flow paths and velocities [Labbas, 2015]. A watershed is an area where all runoff waters converge through the natural drainage system (rivers, streams, lakes), irrigation and drainage systems (canals, ditches) and urban systems (roads, sewer systems...). The artificialization of the watershed surfaces can increase surface runoff at the expense of subsurface runoff, a degradation of the chemical and biological quality of water, a decrease in groundwater recharge and morphological instability of watercourses such as erosion [Bras R.L., 1975; Desbordes, 1989; Labbas, 2015]. The 2022 IPPC report points to soil artificialization as a flood risk factor [Pörtner et al., 2022] while recent studies showed that it could also have more impact on dryness than climate change [Dosdogru et al., 2020]. Moreover, such changes in hydrology could be accompanied by a decrease in biodiversity [Giri et al., 2019]. Finally, [Paudel et al., 2011] argued that the spatial distribution of impermeable surface has a larger influence than the portion of impermeable surfaces within the watershed. Consequences of a flood and a dryness in Toulouse are illustrated in Fig. 1.2.

(a) Picture of La Garonne in Toulouse during the 2022 summer dryness [Balidas et al., 2022]

(b) Picture of La Garonne in Toulouse during a 2014 winter flood [de Fenoyl and Hurtevent, 2014]

Figure 1.2: Pictures of Toulouse during extreme hydrological events

In addition to land artificialization, water management policies have also a strong influence on hydrology. Waste water and storm water are treated by water treatment plants before being discharged into rivers or the sea, unless they are directly diverted by a storm overflow in case of heavy rainfall. Alternative systems such as water tanks, vegetated roofs, porous paving stones or rain gardens, can retain, store, infiltrate or treat part of the rain water [Labbas, 2015]. Therefore, detailed impermeable surfaces maps would be valuable to guide public decision makers towards the most efficient water treatment methods. Finally, it should be noted that two European directives aim to reduce anthropic impacts on watersheds. The

2000 Water Framework Directive sets the objectives in terms of chemical and ecological status of large river basins. The 2007 Flood Directive aims to prevent and limit floods and their consequences on human health, environment and infrastructures.

1.3 Impact of surface sealing on urban micro-climates

Land cover has a significant influence on urban micro-climates [Pörtner et al., 2022] where a significant part of surfaces are impermeable. [Onishi et al., 2010] found a high correlation between the portion of impermeable surfaces and urban heat island (UHI) effects. Urban heat islands occur when ground air temperature in the city center is much higher than ground air temperature outside the city[3]. Materials such as asphalt or tar, which have a low albedo, give back a large part of the solar radiation in the form of heat. In contrast, vegetation cools the environment thanks to the evapotranspiration of leaves, to its high albedo in the infrared and to the shade of trees. In the context of limited resources, impermeable surfaces maps can optimally guide public policies. For instance, [Onishi et al., 2010] studied the mitigation of urban heat islands effects by substituting paved parking lot surfaces with vegetated surfaces in the city of Nagoya, Japan.

1.4 Impact of surface sealing on carbon cycle

Recent scientific studies demonstrated that surface sealing with impermeable surfaces notably affects the soil microbial community and the soil ecosystem [O'Riordan et al., 2021; Pereira et al., 2021]. In particular, surface sealing dramatically decreases the soil carbon storage potential [Commission, 2012; O'Riordan et al., 2021; Pereira et al., 2021; Scalenghe and Marsan, 2009]. As a matter of fact, soil artificialization due to urbanization removes plants and topsoil which can have a significant impact on carbon uptake [Wu et al., 2016] though soils beneath sealed surfaces of pavements and houses can store significant amounts of carbon [Vasenev et al., 2018; Wu et al., 2016; Yan et al., 2015]. For instance, [Yan et al., 2015] estimated that the soil carbon storage of Urumqi, China, decreased by 19% while impermeable surfaces increased by 200% from 1990 to 2010.

2 Mapping artificial impermeable surfaces

In order to better understand the influence of soil artificialization in the context of climate change and to design relevant public policies, scientists and public authorities must be provided with precise maps of artificial impermeable surfaces. To this end, remote sensing offers a great opportunity.

Satellite multispectral images Earth observation satellites have acquired very large amounts of data for the past decades. In particular, the NASA Landsat[4] satellites and the ESA

[3] However, there is no consensus on the threshold at which a UHI occur.
[4] https://landsat.gsfc.nasa.gov/satellites/landsat-8/

Sentinel-2[5] satellites acquire optical images all over the globe with a few-days to a weekly revisit, which allows to produce and update land use and land cover maps. Landsat and Sentinel-2 data are appropriate to map the land cover because of their near infrared (NIR) and short-wave infrared (SWIR) channels complementary to their visible channels (red, green and blue channels), which bring discriminative information on the surfaces composition: such images are called multispectral images.

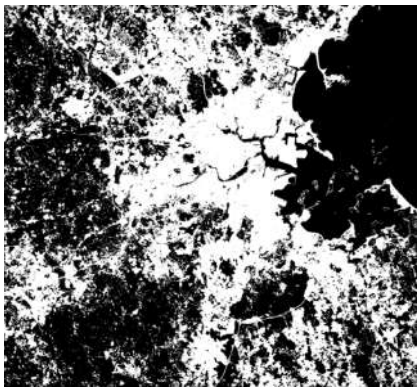
Semantic segmentation In the literature, the production of land cover maps is referred as semantic segmentation [Castillo-Navarro et al., 2021], *i.e.* the process of giving sense to the physics quantities measured by remote sensing sensors (by assigning a land cover class to each pixel). State-of-the-art semantic segmentation techniques optimize machine learning models to correlate input data to land cover classes from a set of pixels annotated with their true classes. This set of pixels is commonly called the **training data set**. The size of the training data set is usually much smaller than the total number of pixels in the image [Castillo-Navarro et al., 2021], which makes the optimization of semantic segmentation models difficult: the challenge is to find **representations** of the data, *i.e.* common features that are shared among data samples gathered under the same semantic class. More importantly, the relations between data representations and semantic classes learned over the training set should hold true on the remaining part of the image: the optimized machine learning model should **generalize** to new data. While many different models could perfectly fit the training data, the assumptions made during the learning process (from the model architecture to the learning algorithm itself), called **inductive biases** [Mitchell, 1980; Zhao et al., 2018], are fundamental to foster models with high generalization capacities.

In recent years, major advances in machine learning have allowed the automatic or semi-automatic mapping of small to medium geographical areas (up to the scale of a country), sometimes supplementing with manual visual inspections of images. Current maps represent the land use of the soils, the land cover of the soils, or both. There are, of course, correlations between the land use and the land cover, as well as correlations with the soils permeability (*e.g.* the surface of roads is mainly impermeable). Therefore, we present both kinds of maps in the following.

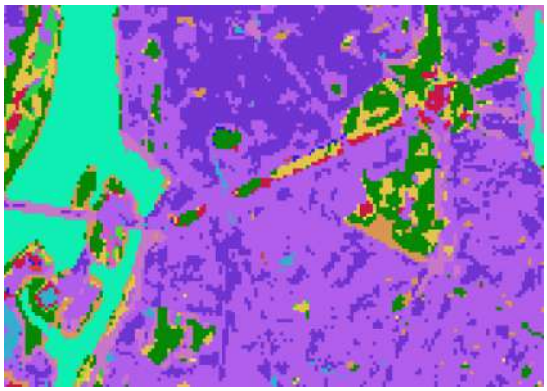
2.1 Available maps of artificial impermeable surfaces

[Sun et al., 2017], [Genyun et al., 2015] and [Xu, 2008] mapped artificial impermeable surfaces against permeable surfaces over the cities of Boston, USA (see Fig. 1.3a), Fuzhou, China and Berlin, Germany, respectively, from Landsat and Sentinel-2 images at a 10 m spatial resolution. The two-classes nomenclature though is reductive of permeability which can take a wide range of values in an urban environment. In contrast, the Imperviousness Density map[6] produced under the Copernicus program[7] of the European Environment Agency provides a degree of impermeability defined as the density of sealed land cover over a 10 m $\times$ 10 m area. The imperviousness map is computed semi-automatically with a machine learning model that uses hand-crafted features derived from Sentinel image time series. However, the imperviousness density map lacks semantic information about the nature of the artificial surfaces, which could for instance be used to derive radiative properties.

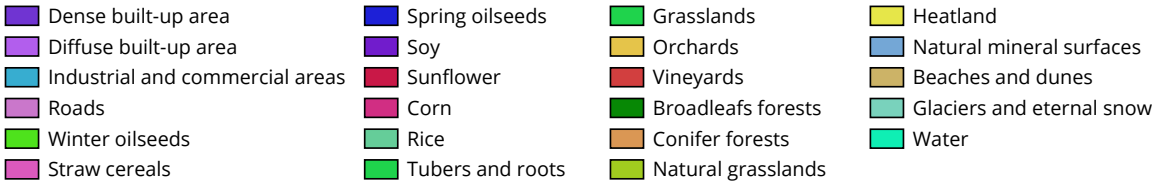
[5] https://www.esa.int/Applications/Observing_the_Earth/Copernicus/Sentinel-2
[6] https://land.copernicus.eu/pan-european/high-resolution-layers/imperviousness
[7] https://www.copernicus.eu/en



(a) Impermeable (white) against permeable (black) map over Boston, USA [Sun et al., 2017]



(b) OSO map over Toulouse city center



(c) OSO nomenclature

Figure 1.3: Examples of land cover maps

2.2 Available land use and land cover maps

CORINE Land Cover[8] is a land use and land cover map over 39 European countries produced by visual inspection in 1990, 2000, 2006, 2012 and 2018 from Landsat and Sentinel-2 (since 2018) images under the Copernicus program. CORINE Land Cover maps 44 classes including 11 classes for artificial surfaces at a 100 m spatial resolution which is obviously not appropriate to urban environments where the characteristic scale is of the order of the meter (*e.g. a tree*). OSO[9] [Inglada et al., 2017; Stoian et al., 2019] is a reduced version of CORINE Land Cover (20 classes approximately over France) produced every year at a 10 m spatial resolution from Sentinel-2 image time series. Meanwhile, Urban Atlas[10] maps have also been produced at a higher spatial resolution under the Copernicus program with a focus on large urban areas. Urban Atlas is a land use map though, hence it cannot be used to study the influence of impermeable surfaces. For instance, it maps the botanic garden in Toulouse city center as a *Green Urban Area* though it is in fact a mix of permeable vegetated, permeable non-vegetated and impermeable surfaces (see Fig. 1.4). In contrast, the OSO map (not specific on urban environments) perceives the diversity of the land cover in the botanic garden, albeit with confusions (see Fig. 1.3b). Higher spatial resolution maps can be produced from airborne images such as the IGN BD Ortho[11]. For instance, impermeable maps of Desclaux and Haillan watersheds were ordered by the Bordeaux regional authority [Bucheli et al., 2015].

To summarize, many land use and land cover maps are publicly available at different scale and spatial resolutions, providing rich semantic information with a multi-class nomenclature. However, those maps mainly describe the land use and land cover with abstract classes such as *Airport*, *Discontinuous urban fabric* or *Forest*, that define conceptual objects. In contrast,

[8] https://land.copernicus.eu/pan-european/corine-land-cover
[9] https://www.theia-land.fr/product/carte-doccupation-des-sols-de-la-france-metropolitaine/
[10] https://land.copernicus.eu/local/urban-atlas
[11] https://geoservices.ign.fr/bdortho

Figure 1.4: Urban Atlas land use map over Toulouse, France.

we would like to map low-abstraction classes such as *Asphalt*, *Tile* or *Tree*, that define the nature of matter.

2.3 Current research efforts to map low-abstraction land cover from airborne hyperspectral images

Airborne hyperspectral images, which have higher spatial and spectral resolutions than spaceborne multispectral images (hundreds of channels in the visible and the infrared spectral domain instead of common RGB channels), offer a great opportunity to produce low-abstraction land cover maps.

Spectral intra-class variability The challenge of semantic segmentation of hyperspectral images is to design and optimize machine learning models that capture the spectral intra-class variability [Adep et al., 2017; Chang, 2000; Xue et al., 2020] in the context of small training data sets. Spectral intra-class variability describes the variation of spectral information between different pixels that belong to the same semantic class. Intra-class variability has different causes: variations in illumination conditions, slight variations in the material composition (for instance, variations in water or chlorophyll content of different trees from the same specie or variations of tar due to aging) and from larger variations in the material composition, due to the fact that the nomenclature gathers different materials under the same class (for instance, different tree species gathered in a unique class)[Revel et al., 2018]. For convenience, we respectively name those different kinds of variability as *physics* , *intrinsic* and *semantic* intra-class variability.

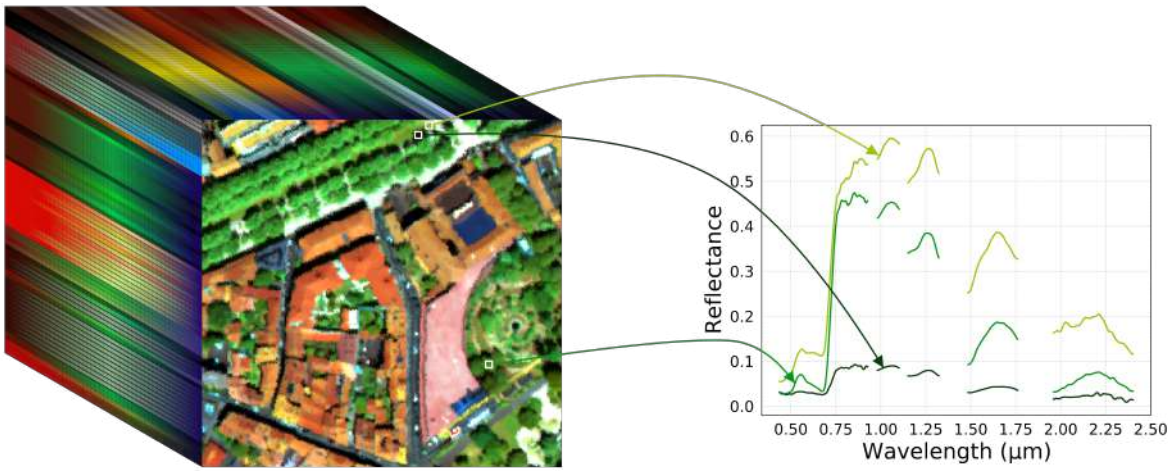


Figure 1.5: Illustration of a hyperspectral image and three different spectra of vegetation

Field campaigns Labeling the nature of soil materials (*e.g.* discriminating asphalt from gravels) by photo interpretation is much more difficult than labeling abstract classes such as roads or roofs. Thus, expensive and time consuming field campaigns, in addition to photo interpretation, are necessary to annotate images, which considerably limits the size of training data sets.

In the past years, extensive research efforts have been conducted to leverage the information of airborne hyperspectral data for semantic segmentation with few labeled samples, which mainly followed **three independent research directions**: 1) *the architecture of machine learning models*, 2) *the quality of the training data sets* and 3) *the optimization algorithms*. Although these lines of research focus on different levels, they have the same objective: **improving the representation of data to handle intra-class variability**.

Architecture of machine learning models Random forests [Breiman, 2001] and kernel methods such as support vector machine [Cortes and Vapnik, 1995] have been widely applied on hyperspectral data due to their ability to handle their high spectral dimension. Although, from the past years, spatial-spectral convolutional neural networks (3D CNNs), which learn spatial and spectral dependencies, have become the state of the art among supervised techniques [Audebert et al., 2019; Rasti et al., 2020]. For instance, [Li et al., 2017] introduced a 3D CNN that reached very high accuracy on small public hyperspectral data sets: Pavia University[12], Indian Pines[12] and Botswana[12]. However, they did not experiment their model on larger and more complex images with a higher number of classes and higher intra-class variability.

Quality of the training data set Active learning techniques aim to improve the training data set by interactively and iteratively adding a few labeled pixels, through the interplay of queries given by a machine learning algorithm and annotations given by an expert [Settles, 2012]. [Di and Crawford, 2010; Li et al., 2011; Rajan et al., 2008] introduced different active learning strategies in the context of hyperspectral image semantic segmentation. They empirically demonstrated the benefits of active learning in term of semantic segmentation accuracy on small public hyperspectral images, showing that labeling only a few additional pixels can have a significant impact. However, they did not demonstrate the benefits of active learning methods in a real use case where three difficulties arise: the number of unlabeled pixels (*i.e.*

[12] https://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes

the number of candidates) is considerably larger, the initial nomenclature is usually not exhaustive enough to represent the diversity of the land cover and the intra-class variability is much larger.

Optimization algorithms In order to improve the optimization of machine learning models in the context of small training data sets, various auxiliary losses have been introduced to leverage the structure of unlabeled data [Hendrycks et al., 2019; Laine and Aila, 2016] which have been used and adapted to hyperspectral semantic segmentation [Liu et al., 2017; Rasti et al., 2020; Yue et al., 2021]. Experimental results on small public hyperspectral data sets showed that unsupervised or semi-supervised learning can outperform common supervised optimization algorithms in term of segmentation accuracy. However, the potential of semi-supervised algorithms has not been studied on large scale hyperspectral scenes, where many unlabeled pixels do not belong to a class of the labeled training set.

2.4 Limitations of available maps

Table 1.1: Characteristics of available land cover and land use maps. Maps that are specifically over urban areas are shown in bold.

Maps	**Nomenclature**	**Number of classes**	**Abstraction**	**Scale**	**Spatial resolution**
CORINE Land Cover[8]	Land cover & land use	> 40	+++	+++	100 m
Urban Atlas[10]	Land use & land cover	> 20	+++	++	10 m
OSO[9]	Land cover	> 20	+	++	10 m
Imperviousness density map[6]	Permeability	100	- - -	++	10 m
Satellite-derived impermeable maps [11]	Permeable Impermeable	2	-	+	10 m
Airborne-derived impermeable maps [12]	Vegetation Building Impermeable	3	-	+	< 1 m
Airborne hyperspectral derived maps [13]	Land cover	$[10, 20]$	- - -	- - -	$[1m, 20m]$
Ideal impermeable map	Land cover	> 40	- - -	+++	< 1 m

[8] https://land.copernicus.eu/pan-european/corine-land-cover
[10] https://land.copernicus.eu/local/urban-atlas
[9] https://www.theia-land.fr/product/carte-doccupation-des-sols-de-la-france-metropolitaine/
[6] https://land.copernicus.eu/pan-european/high-resolution-layers/imperviousness/
[11] [Genyun et al., 2015; Sun et al., 2017; Xu, 2008]
[12] [Bucheli et al., 2015]
[13] [Audebert et al., 2019; Rasti et al., 2020]

To summarize, state-of-the-art semantic segmentation techniques can:

- automatically or semi-automatically map abstract land cover and land use classes at large scale and medium spatial resolution from multispectral satellite images,
- automatically map abstract land cover and land use classes of a small geographical area at high spatial resolution from RGB airborne images,
- automatically map soil materials of a very small geographical area at high spatial resolution from airborne hyperspectal images.

Nevertheless, current semantic segmentation methods have not demonstrated their ability to produce land cover maps with a large number of classes, at high spatial resolution and over large areas (typically over a metropolis) from airborne hyperspectral images. The characteristics of available maps are summarized in Tab. 1.1 and are contrasted with the characteristics of an ideal artificial impermeable map.

3 Objectives of the thesis

Given the current limitations in hyperspectral image analysis, our general objective is to develop a methodology to automatically and exhaustively map the artificial impermeable surfaces of a metropolis from airborne hyperspectral images. More specifically, we aim to address the scarcity of ground truth annotations by exploiting unlabeled data from three different perspectives: 1) by labeling a very small subset of unlabeled data, 2) by integrating a priori physical knowledge into semi-supervised learning and 3) by learning class-specific spectral features without the supervision of class labels.

■ First objective

In consequence of the cost and difficulty to annotate pixels with land cover classes, labeled spectra are generally very unrepresentative of the true intra-class variability. Therefore, no matter how sophisticated the semantic segmentation models are, they will not be able to produce high quality maps from low quality training data. Besides, defining an appropriate nomenclature for a land cover map of a metropolis is not trivial because of the great diversity of materials. Inevitably, materials missing from the nomenclature will be wrongly predicted. In this context, Active Learning (AL), which aims to find the best samples to label, is a very promising research direction. Until now, scientific studies on AL for hyperpsectral image analysis have not considered realistic scenarios on large-scale images for which exploration (*i.e.* finding new land cover materials than those in the initial training set) is critical. **Therefore, our first objective is to study the possibility to significantly improve the mapping of impermeable surfaces over a metropolis from a hyperspectral image by labeling only a few pixels.** Among the hundreds of millions of pixels in the image, can AL algorithms select a very small subset to label so that it jointly improves the representativeness of the nomenclature and the machine learning models capacities to capture the intra-class variability? To answer these questions, we select seven state-of-the-art AL techniques that showcase different strategies and introduce an experimental plan to study their complementarity. In addition, we introduce a preprocessing technique to handle large-scale images that is essential to use AL methods with significant computational require-

ments. Finally, we enhance a state-of-the-art AL technique by integrating a priori semantic knowledge (derived from the permeability of materials) to specifically address the semantic segmentation of impermeable surfaces.

■ **Second objective**

Although active learning seems imperative on large-scale images, the additional training data would unlikely be sufficient to perfectly capture the spectral intra-class variability. Active learning techniques are indeed limited by the capacity of semantic segmentation models (that they use themselves) to learn discriminative representations. It is, therefore, a vicious circle: machine learning algorithms need informative training data to induce discriminative data representations and active learning algorithms need well-trained machine learning models to select informative data. Thus, we can wonder whether unlabeled data could bring useful information to representation learning. **Precisely, our second objective is to guide representation learning from labeled and unlabeled data with deductive physics-based biases, *i.e.* assumptions derived from physical prior knowledge.** As a matter of fact, the spectral variations caused by local changes in illumination are ruled by physical laws that obviously apply similarly on labeled and unlabeled data. To jointly leverage our physical prior knowledge and the structure of unlabeled data, we develop and validate a semi-supervised hybrid model, which is the combination of a machine learning model and a physics model, on simulated and real hyperspectral images.

■ **Third objective**

If the combination of physics and machine learning models would perfectly capture the *physics* intra-class variability, the representation of the *intrinsic* and *semantic* intra-class variabilities would still be limited by the number of additional pixels active learning techniques have at their disposal. Accordingly, could we leverage unlabeled data to learn representations robust to *intrinsic* and *semantic* variations? While state-of-the-art unsupervised techniques have demonstrated impressive results in the machine learning community to learn high-level representations, *i.e.* features that are specific to abstract concepts and robust to contextual intra-class variations, *intrinsic* and *semantic* variabilities induce low-level spectral variations that mostly depend on the chemical composition of matter. Consequently, the key issue is to learn a representation space in which data samples are grouped according to their land cover class. **Therefore, our third objective is to improve the discriminative potential of spectral representations without supervision.** For that purpose, we focus on autoencoding-based techniques and empirically study their potential on a new hyperspectral data set that allows a fair evaluation of unsupervised methods thanks to its size and standard training and test sets.

4 Organization of the manuscript

In Chapter 2, we present the state of the art in semantic segmentation, on which our research is based. In Chapter 3, we present the materials and methods that were introduced by the hyperspectral community to benchmark segmentation models as well as hyperspectral data that were provided by ONERA. Chapters 4, 5 and 6 are the core of our contributions. We

conclude our work and discuss perspectives in Chapter 7.

5 References

Adep, R. N., Ramesh, H., et al. (2017). Exhype: A tool for mineral classification using hyperspectral data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 124:106–118. 19

Audebert, N., Le Saux, B., and Lefèvre, S. (2019). Deep learning for classification of hyperspectral data: A comparative review. *IEEE geoscience and remote sensing magazine*, 7(2):159–173. 20, 21

Balidas, J., Occitanie, F. B., and Bleu, F. (2022). Frace bleu, sécheresse : les photos impressionnantes du niveau de la garonne à toulouse. https://www.francebleu.fr/infos/environnement/secheresse-les-photos-impressionnantes-du-niveau-de-la-garonne-a-toulouse-1659619430. 15

Bras R.L., P. F. (1975). Effects of urbanization on catchment response. *Journal of Hydraulics Division*, 101:451–466. 15

Breiman, L. (2001). Random forests. *Machine learning*, 45:5–32. 20

Bucheli, H., Goussot, E., and Gonzalez-Alvarez, A. (2015). Maîtriser l'imperméabilisation des sols - enjeux et méthodes. https://www.aurba.org/wp-content/uploads/2015/06/Impermabilisation_des_sols.pdf. 18, 21

Castillo-Navarro, J., Le Saux, B., Boulch, A., Audebert, N., and Lefèvre, S. (2021). Semi-supervised semantic segmentation in earth observation: The minifrance suite, dataset analysis and multi-task network study. *Machine Learning*, pages 1–36. 17

CEV, C. p. l. V. (2019). Les enjeux de l'artificialisation des sols : diagnostic. https://www.ecologie.gouv.fr/sites/default/files/Les enjeux de l'artificialisation des sols.pdf. 13, 15

Chang, C.-I. (2000). An information-theoretic approach to spectral variability, similarity, and discrimination for hyperspectral image analysis. *IEEE Transactions on information theory*, 46(5):1927–1932. 19

Commission, E. E. (2012). Guidelines on best practice to limit, mitigate or compensate soil sealing. *Luxembourg: European Union SWD (2012) 101*. 16

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20:273–297. 20

de Fenoyl, X. and Hurtevent, X. (2014). Toulouse : alerte aux inondations. https://www.ladepeche.fr/article/2014/01/25/1803137-toulouse-alerte-aux-inondations.html. 15

Desbordes, M. (1989). Principales causes d'aggravation des dommages dus aux inondations par ruissellement superficiel en milieu urbanisé. *Bulletin hydrologie urbaine*, 4:2–10. 15

Di, W. and Crawford, M. M. (2010). Multi-view adaptive disagreement based active learning for hyperspectral image classification. In *2010 IEEE International Geoscience and Remote Sensing Symposium*, pages 1374–1377. 20

Dosdogru, F., Kalin, L., Wang, R., and Yen, H. (2020). Potential impacts of land use/cover and climate changes on ecologically relevant flows. *Journal of Hydrology*, 584:124654. 13, 15

Genyun, S., Chen, X., Jia, X., Yao, Y., and Wang, Z. (2015). Combinational build-up index (cbi) for effective impervious surface mapping in urban areas. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 9:1–12. 17, 21

Giri, S., Zhang, Z., Krasnuk, D., and Lathrop, R. G. (2019). Evaluating the impact of land uses on stream integrity using machine learning algorithms. *Science of The Total Environment*, 696:133858. 13, 15

Hendrycks, D., Mazeika, M., Kadavath, S., and Song, D. (2019). Using self-supervised learning can improve model robustness and uncertainty. *Advances in neural information processing systems*, 32. 21

Inglada, J., Vincent, A., Arias, M., Tardy, B., Morin, D., and Rodes, I. (2017). Operational high resolution land cover map production at the country scale using satellite image time series. *Remote Sensing*, 9(1):95. 18

Labbas, M. (2015). *Modélisation hydrologique de bassins versants périurbains et influence de l'occupation du sol et de la gestion des eaux pluviales. Application au bassin de l'Yzeron (130 km2)*. Theses, Doctorat, spécialité : Océan, Atmosphère, Hydrologie, Université de Grenoble. 13, 14, 15

Laine, S. and Aila, T. (2016). Temporal ensembling for semi-supervised learning. In *International Conference on Learning Representations*. 21

Li, J., Bioucas-Dias, J. M., and Plaza, A. (2011). Hyperspectral image segmentation using a new bayesian approach with active learning. *IEEE Transactions on Geoscience and Remote Sensing*, 49(10):3947–3960. 20

Li, Y., Zhang, H., and Shen, Q. (2017). Spectral–spatial classification of hyperspectral imagery with 3d convolutional neural network. *Remote Sensing*, 9(1):67. 20

Liu, B., Yu, X., Zhang, P., Tan, X., Yu, A., and Xue, Z. (2017). A semi-supervised convolutional neural network for hyperspectral image classification. *Remote Sensing Letters*, 8(9):839–848. 21

Magnan, J.-P. (1999). L'eau dans le sol. *Techniques de l'ingénieur Géotechnique*, base documentaire : TIB238DUO.(ref. article : c212). 14

Mitchell, T. M. (1980). *The need for biases in learning generalizations*. 17

Onishi, A., Cao, X., Ito, T., Shi, F., and Imura, H. (2010). Evaluating the potential for urban heat-island mitigation by greening parking lots. *Urban Forestry and Urban Greening*, 9(4):323 – 332. 16

O'Riordan, R., Davies, J., Stevens, C., and Quinton, J. N. (2021). The effects of sealing on urban soil carbon and nutrients. *SOIL*, 7(2):661–675. 16

Paudel, M., Nelson, E. J., Downer, C. W., and Hotchkiss, R. (2011). Comparing the capability of distributedand lumped hydrologic models for analyzing the effects of land use change. *Journal of Hydroinformatics*, 13(3):461–473. 15

Pereira, M. C., O'Riordan, R., and Stevens, C. (2021). Urban soil microbial community and microbial-related carbon storage are severely limited by sealing. *Journal of Soils and Sediments*, 21:1455–1465. 16

Pörtner, H.-., Roberts, D., Tignor, M., E.S., P., Mintenbeck, K., Alegría, A., Craig, S., Langsdorf, S., Löschke, S., Möller, V., Okem, A., Rama, B., and eds (2022). *Climate Change 2022: Impacts, Adaptation and Vulnerability. Contribution of Working Group II to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change*, volume 1. Cambridge University Press. 13, 15, 16

Rajan, S., Ghosh, J., and Crawford, M. M. (2008). An active learning approach to hyperspectral data classification. *IEEE Transactions on Geoscience and Remote Sensing*, 46(4):1231–1242. 20

Rasti, B., Hong, D., Hang, R., Ghamisi, P., Kang, X., Chanussot, J., and Benediktsson, J. A. (2020). Feature extraction for hyperspectral imagery: The evolution from shallow to deep: Overview and toolbox. *IEEE Geoscience and Remote Sensing Magazine*, 8(4):60–88. 20, 21

Revel, C., Deville, Y., Achard, V., Briottet, X., and Weber, C. (2018). Inertia-constrained pixel-by-pixel nonnegative matrix factorisation: A hyperspectral unmixing method dealing with intra-class variability. *Remote Sensing*, 10(11):1706. 19

Scalenghe, R. and Marsan, F. A. (2009). The anthropogenic sealing of soils in urban areas. *Landscape and urban planning*, 90(1-2):1–10. 16

Settles, B. (2012). *Active Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan and Claypool Publishers. 20

Stoian, A., Poulain, V., Inglada, J., Poughon, V., and Derksen, D. (2019). Land cover maps production with high resolution satellite image time series and convolutional neural networks: Adaptations and limits for operational systems. *Remote Sensing*, 11(17):1986. 18

Strand, G.-H. (2022). Accuracy of the copernicus high-resolution layer imperviousness density (hrl imd) assessed by point sampling within pixels. *Remote Sensing*, 14(15):3589. 13

Sun, Z., Du, W., Jiang, H., Weng, Q., Guo, H., Han, Y., Xing, Q., and Ma, Y. (2022). Global 10-m impervious surface area mapping: A big earth data based extraction and updating approach. *International Journal of Applied Earth Observation and Geoinformation*, 109:102800. 15

Sun, Z., Wang, C., Guo, H., and Shang, R. (2017). A modified normalized difference impervious surface index (mndisi) for automatic urban mapping from landsat imagery. *Remote Sensing*, 9:942. 17, 18, 21

Vasenev, V., Stoorvogel, J., Leemans, R., Valentini, R., and Hajiaghayeva, R. (2018). Projection of urban expansion and related changes in soil carbon stocks in the moscow region. *Journal of Cleaner Production*, 170:902–914. 16

Wu, X., Hu, D., Ma, S., Zhang, X., Guo, Z., and Gaston, K. J. (2016). Elevated soil co2 efflux at the boundaries between impervious surfaces and urban greenspaces. *Atmospheric Environment*, 141:375–378. 16

Xu, H. (2008). A new index for delineating built-up land features in satellite imagery. *International Journal of Remote Sensing*, 29:4269 – 4276. 17, 21

Xue, Y., Zeng, D., Chen, F., Wang, Y., and Zhang, Z. (2020). A new dataset and deep residual spectral spatial network for hyperspectral image classification. *Symmetry*, 12(4):561. 19

Yan, Y., Zhang, C., Hu, Y., and Kuang, W. (2015). Urban land-cover change and its impact on the ecosystem carbon storage in a dryland city. *Remote Sensing*, 8(1):6. 16

Yue, J., Fang, L., Rahmani, H., and Ghamisi, P. (2021). Self-supervised learning with adaptive distillation for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–13. 21

Zhao, S., Ren, H., Yuan, A., Song, J., Goodman, N., and Ermon, S. (2018). Bias and generalization in deep generative models: An empirical study. *Advances in Neural Information Processing Systems*, 31. 17

Chapter 2

Semantic segmentation: state of the art

Contents

This chapter presents the machine learning fundamentals and state-of-the-art semantic segmentation techniques on which this thesis is based. Firstly, we present in section 1 the basics of statistical models and how they can be used to make predictions. Secondly, we present in section 2 the state-of-the-art techniques to extract features from data. Finally, we present in section 3 how statistical models are optimized.

1 Statistical modeling

Images are high-dimensional data, and hyperspectral images even more so. Consider for instance a hyperspectral image of 32×32 pixels with 100 spectral channels: the image lies in a space with $32 \times 32 \times 100 = 102,400$ dimensions. If the image is encoded on 16 bits, then the volume of the data space, or equivalently the number of different possible images, is equal to $2^{16\times32\times32\times100} = 2^{1,638,400}$. A spectrum alone (*i.e.* one pixel only) could take 2^{1600} different values, which is 10^{400} times more than the estimated number of atoms in the observable universe [Vopson, 2021]. In practice, however, data usually occupies a very small subset of the whole data space [Goodfellow et al., 2016]. For instance, hyperspectral data are continuous over the spectral dimension, which forbids, de facto, unrealistic regions of the data space that are not consistent with spectral continuity. Therefore, it is natural to model data as a random variable X that takes values in $\mathcal{X}$, which density probability function $p(X)$ indicates how likely are some regions of the data space.

For classification tasks, there are two approaches, namely generative modeling and discriminative modeling [Jebara, 2012]. Generative models aim to approximate the joint probability distribution $p(X, Y)$ of the data X and the semantic class Y, which is a discrete random variable taking integer values in $\{1, \dots, c\}$, c being the number of classes. The categorical conditional distribution of Y given X, denoted as $p(Y|X)$, is then computed from Bayes rule. In contrast, discriminative models directly approximate the conditional distribution $p(Y|X)$.

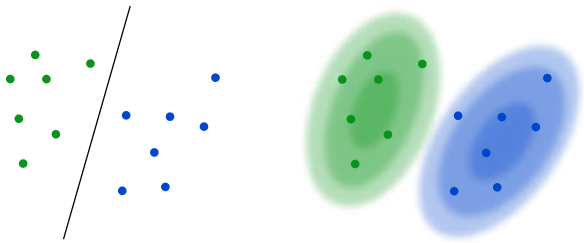


Figure 2.1: Illustration of the discriminative and generative paradigms

In section 1.1, we briefly introduce the concepts of probability theory that are fundamental to machine learning. In sections 1.2 and 1.3, we present discriminative and generative modeling, respectively.

1.1 Probability theory: fundamentals

In this section, we wish to give a brief and intuitive introduction to the fundamental notions of probability theory that will be extensively used in machine learning. For a more exhaustive and rigorous introduction, we suggest to the reader the following text books: [Loève, 1977; Murphy, 2012].

Continuous random variables. Let X be a continuous variable that randomly takes values in $\mathcal{X}$. If X is multivariate, *i.e.* $|\mathcal{X}| > 1$, we denote realizations of X as $\boldsymbol{x}$, and x otherwise. By default, we assume that X is multivariate.

The density $p : \mathcal{X} \longrightarrow \mathbb{R}^+$ defines a probability distribution if $\int_{\mathcal{X}} p(X = \boldsymbol{x})d\boldsymbol{x} = 1$. Then, the probability of X taking values in the subset $\mathcal{A} \subset \mathcal{X}$ is given by $\int_{\mathcal{A}} p(X = \boldsymbol{x})d\boldsymbol{x}$.

Discrete random variables. Let Y be a discrete variable that randomly takes values in $\mathcal{Y}$. The mass function $p : \mathcal{Y} \longrightarrow [0, 1]$ defines a probability distribution over Y if $\sum_y p(Y = y) =$

1. Then, the probability that $Y = y$ is directly given by the mass function $p(Y = y)$.

In the following, we make the common abuse of notation which consists in confusing the realizations of the random variable and the random variable itself, denoting the random variable as $\boldsymbol{x}$ and the density probability function as $p(\boldsymbol{x})$.

Expectation. The expectation of some function f with respect to a probability distribution $p(x)$ is the average f takes for random realizations of $\boldsymbol{x}$. We denote it as $\mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x})}[f(\boldsymbol{x})]$ (the expected value of a random variable $\boldsymbol{x}$ is simply denoted as $\mathbb{E}[\boldsymbol{x}]$) and is computed as follows:

- for continuous random variables:

$$\mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x})}[f(\boldsymbol{x})] = \int p(\boldsymbol{x}) f(\boldsymbol{x}) d\boldsymbol{x} \tag{2.1}$$

- for discrete random variables:

$$\mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x})}[f(\boldsymbol{x})] = \sum_{x} p(\boldsymbol{x}) f(\boldsymbol{x}) \tag{2.2}$$

For continuous random variables, if the expectation cannot be computed analytically, it can be computed numerically with Monte-Carlo sampling:

$$\mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x})}[f(\boldsymbol{x})] \approx \frac{1}{K} \sum_{k=1}^{K} f(\boldsymbol{x}^{(k)}) \; ; \; \boldsymbol{x}^{(k)} \sim p(\boldsymbol{x}) \tag{2.3}$$

Similarly, if we have a finite number of data points $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\}$ independently and identically sampled from an unknown distribution $p_{data}(\boldsymbol{x})$, the expectation can be approximated as follows:

$$\mathbb{E}_{\boldsymbol{x}\sim p_{data}(\boldsymbol{x})}[f(\boldsymbol{x})] \approx \frac{1}{N} \sum_{i=1}^{N} f(\boldsymbol{x}^{(i)}) = \mathbb{E}_{\boldsymbol{x}\sim \hat{p}_{data}(\boldsymbol{x})}[f(\boldsymbol{x})] \tag{2.4}$$

where $\hat{p}_{data}(\boldsymbol{x})$ is the empirical data distribution that assigns a $\frac{1}{N}$ probability to each sample. Fig. 2.2 illustrates how we can approximate the expectation of a random variable whose true density is unknown but from which we can draw samples.

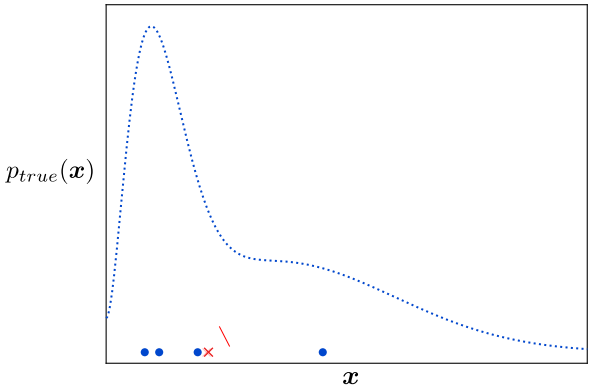


Figure 2.2: Illustration of Monte-Carlo sampling with a true, yet unknown, density plotted in the blue dotted line. Monte-Carlo samples are shown as blue points and the estimation of the expectation is shown as a red cross.

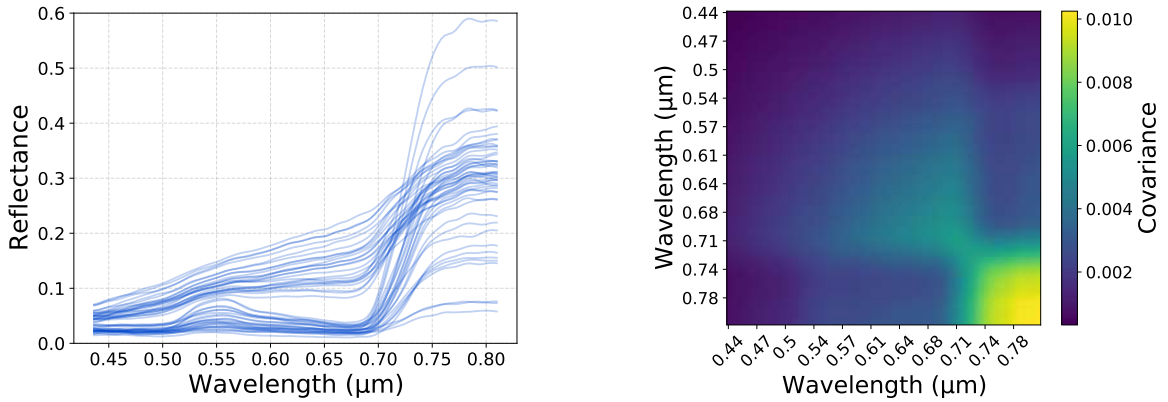
Covariance. The covariance of a multivariate random variable $\boldsymbol{x}$, denoted as $\boldsymbol{\Sigma}_{\boldsymbol{x}}$, expresses how different coordinates of the random variable relate to each other. It is defined as follows:

$$\boldsymbol{\Sigma}_{\boldsymbol{x}} = \mathbb{E}[(\boldsymbol{x} - \mathbb{E}[\boldsymbol{x}])(\boldsymbol{x} - \mathbb{E}[\boldsymbol{x}])^T] \tag{2.5}$$

The i^{th} diagonal element of $\mathbf{\Sigma}_{\boldsymbol{x}}$ is the variance of the i^{th} coordinate of $\boldsymbol{x}$. For data points $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\}$, the empirical covariance approximates the true covariance:

$$\mathbf{\Sigma}_{\boldsymbol{x}}^{emp} = \frac{1}{N} \sum_{i=1}^{N} (\boldsymbol{x}^{(i)} - \bar{\boldsymbol{x}})(\boldsymbol{x}^{(i)} - \bar{\boldsymbol{x}})^T \tag{2.6}$$

where $\bar{\boldsymbol{x}}$ denotes the mean of the data points.



(a) Vegetation spectra

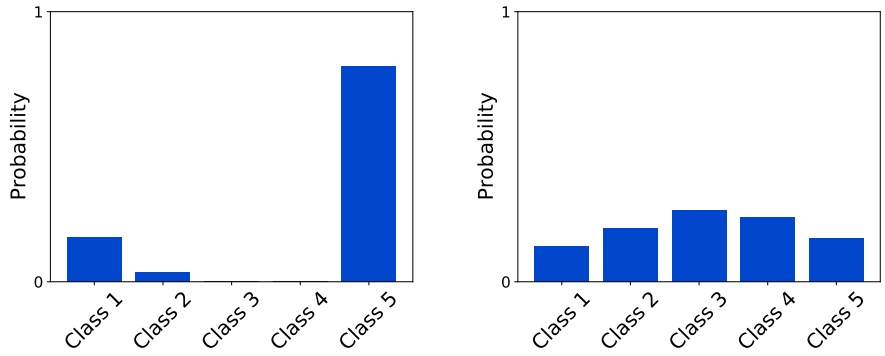
(b) Empirical covariance

Figure 2.3: Illustration of an empirical covariance computed from samples of vegetation spectra

Entropy. The entropy of a discrete random variable y, denoted as $H(y)$, is defined as follows:

$$H(y) = \mathbb{E}_{y \sim p(y)} \frac{1}{\log\, p(y)} = -\sum_{y} p(y) \log\, p(y) \tag{2.7}$$

Entropy can be understood as a measure of uncertainty. Fig. 2.4 illustrates the mass function of random variables with low and high entropy.



(a) Low-entropy random variable

(b) High-entropy random variable

Figure 2.4: Probability distributions of discrete random variables over some classes. When the mass is mainly distributed on one class only, the entropy is low. On the contrary, when the mass is homogeneously spread over the different classes, the entropy is high.

Kullback-Leibler divergence. The Kullback-Leibler (KL) divergence measures how two distributions $p(\boldsymbol{x})$ and $q(\boldsymbol{x})$ look alike (illustrated in Fig. 2.5):

$$D_{KL}(p(\boldsymbol{x})||q(\boldsymbol{x})) = \mathbb{E}_{\boldsymbol{x} \sim p(\boldsymbol{x})}[\log \frac{p(\boldsymbol{x})}{q(\boldsymbol{x})}] = \int p(\boldsymbol{x}) \log \frac{p(\boldsymbol{x})}{q(\boldsymbol{x})} d\boldsymbol{x} \tag{2.8}$$

In particular, the KL divergence has the following properties:

- $D_{KL}(p(\boldsymbol{x})||q(\boldsymbol{x})) \geq 0$,

- If $p(\boldsymbol{x}) = q(\boldsymbol{x})$ for all $\boldsymbol{x}$, then $D_{KL}(p(\boldsymbol{x})||q(\boldsymbol{x})) = 0$,
- $D_{KL}(p(\boldsymbol{x})||q(\boldsymbol{x})) \neq D_{KL}(q(\boldsymbol{x})||p(\boldsymbol{x}))$.

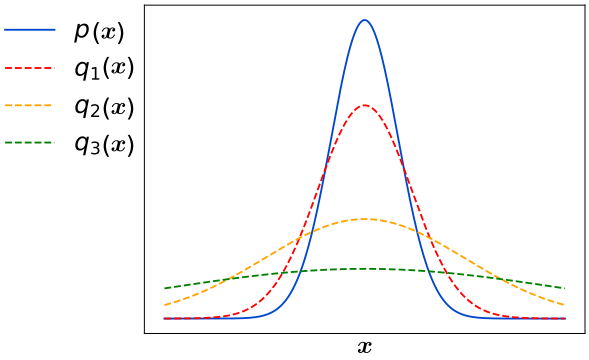


Figure 2.5: $D_{KL}(p(\boldsymbol{x})||q_1(\boldsymbol{x})) \leq D_{KL}(p(\boldsymbol{x})||q_2(\boldsymbol{x})) \leq D_{KL}(p(\boldsymbol{x})||q_3(\boldsymbol{x}))$

Density estimation. As we have seen, generative and discriminative modeling both aim to approximate some unknown data distributions. A perfect estimation of a density $p_{data}(\boldsymbol{x})$, respectively $p_{data}(y|\boldsymbol{x})$), by a parametric distribution $p_{\boldsymbol{\theta}}(\boldsymbol{x})$, respectively $p_{\boldsymbol{\theta}}(y|\boldsymbol{x})$, should minimize the KL divergence between the true and the approximated distributions $D_{KL}(p_{data}(\boldsymbol{x})||p_{\boldsymbol{\theta}}(\boldsymbol{x}))$, respectively $D_{KL}(p_{data}(y|\boldsymbol{x})||p_{\boldsymbol{\theta}}(y|\boldsymbol{x}))$. If we consider a generative problem for the sake of simplicity, a natural idea to find the best approximation $p_{\boldsymbol{\theta}^*}(\boldsymbol{x})$ is therefore to minimize the KL divergence between the empirical data distribution (that puts $1/N$ probability on every points $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\}$ of the training data set $\mathcal{D}$) and the approximated distribution:

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} D_{KL}(\hat{p}_{data}(\boldsymbol{x})||p_{\boldsymbol{\theta}}(\boldsymbol{x})) \tag{2.9}$$

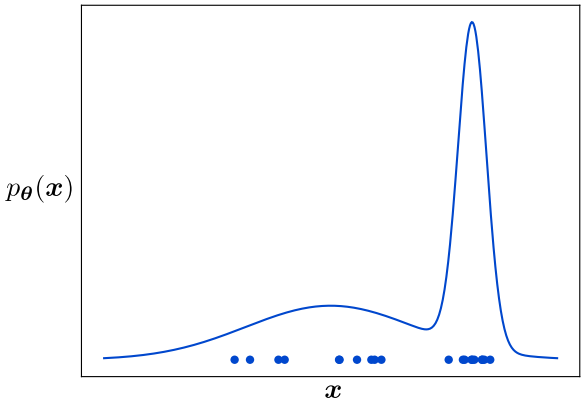


Figure 2.6: Illustration of an approximated probability distribution, modeled as a mixture of gaussians, that maximizes the likelihood of data samples.

In fact, minimizing this KL divergence is equivalent to maximizing the likelihood of the training data [Goodfellow et al., 2016], which is called maximum likelihood estimation (MLE) in the literature:

$$\begin{aligned}
\boldsymbol{\theta}^* &= \arg\min_{\boldsymbol{\theta}} D_{KL}(\hat{p}_{data}(\boldsymbol{x})||p_{\boldsymbol{\theta}}(\boldsymbol{x})) && (2.10)\\
&= \arg\min_{\boldsymbol{\theta}} \mathbb{E}_{\boldsymbol{x}\sim\hat{p}_{data}(\boldsymbol{x})}[\log \hat{p}_{data}(\boldsymbol{x}) - \log p_{\boldsymbol{\theta}}(\boldsymbol{x})] && (2.11)\\
&= \arg\min_{\boldsymbol{\theta}} -\mathbb{E}_{\boldsymbol{x}\sim\hat{p}_{data}(\boldsymbol{x})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x})] && (2.12)\\
&= \arg\min_{\boldsymbol{\theta}} -\sum_{\boldsymbol{x}^{(i)}\in\mathcal{D}} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)}) && (2.13)\\
&= \arg\max_{\boldsymbol{\theta}} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}) && (2.14)
\end{aligned}$$

1.2 Discriminative modeling

In the following, we consider a training data set $\mathcal{D}_{train}$ composed of N data points and their associated semantic classes $\{(\boldsymbol{x}^{(1)}, y^{(1)}), \ldots, (\boldsymbol{x}^{(N)}, y^{(N)})\}$ independently and identically sampled from an unknown distribution $p_{data}(\boldsymbol{x}, y)$ that takes values in $\mathcal{X} \times \{1, \ldots, c\}$. The objective of discriminative models is to approximate the conditional distribution $p_{data}(y|\boldsymbol{x})$, also called the likelihood. We denote the approximation of the likelihood as $p_{\boldsymbol{\theta}}(y|\boldsymbol{x})$:

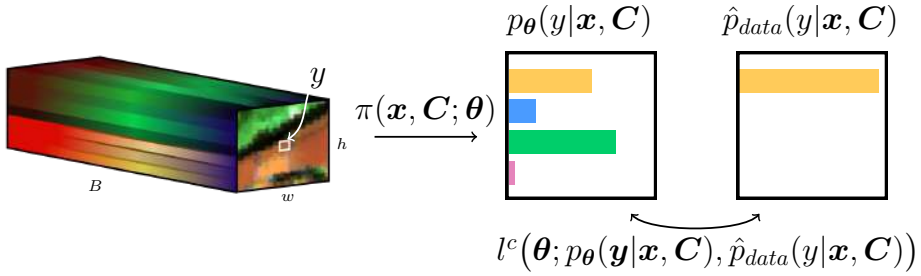
$$\forall i \in \{1, \ldots, c\},\ p_{\boldsymbol{\theta}}(y = i|\boldsymbol{x}) = \pi_i(\boldsymbol{x}; \boldsymbol{\theta}) \text{ s.t. } 0 \leq \pi_i(x; \boldsymbol{\theta}) \leq 1\ ;\ \sum_{i=1}^{c} \pi_i(x; \boldsymbol{\theta}) = 1 \tag{2.15}$$

where the mapping $\pi : \mathcal{X} \longrightarrow [0, 1]^c$ is parameterized by $\boldsymbol{\theta}$. The optimal parameters $\boldsymbol{\theta}^*$ minimize the KL divergence between the approximated distribution and the empirical distribution of the data $\hat{p}_{data}(y, |\boldsymbol{x})$, which is equivalent to the data negative log marginal likelihood, as seen in section 1.1. Thus, for a data point $(\boldsymbol{x}^{(i)}, y^{(i)})$, we denote both losses as $l^c(\boldsymbol{\theta}; p_{\boldsymbol{\theta}}(y^{(i)}|\boldsymbol{x}^{(i)}), \hat{p}_{data}(y^{(i)}|\boldsymbol{x}^{(i)}))$ which leads to the following total loss function:

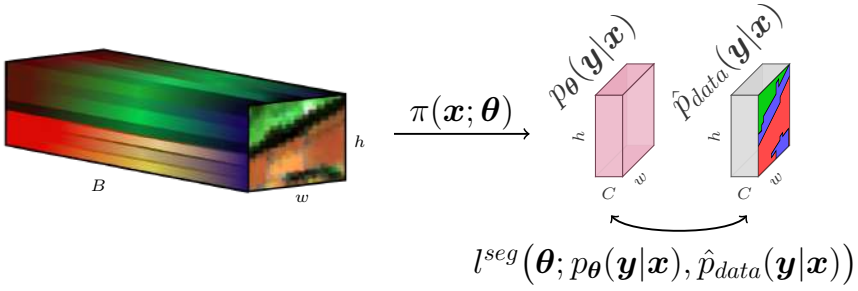
$$\mathcal{L}^c(\boldsymbol{\theta}; \mathcal{D}_{train}) = \frac{1}{N} \sum_{i=1}^{N} l^c(\boldsymbol{\theta}; p_{\boldsymbol{\theta}}(y^{(i)}|\boldsymbol{x}^{(i)}), \hat{p}_{data}(y^{(i)}, |\boldsymbol{x}^{(i)})) \tag{2.16}$$

Discriminative modeling for hyperspectral image segmentation can be viewed from two perspectives: pixel classification and patch segmentation.

Pixel classification. Because the spectral dimension of hyperspectral images is very informative, image segmentation can be performed by classifying the pixels of the image individually, independently of the others. Most of the time, however, the neighborhood of the pixel is considered to benefit from the spatial information. Thus, the task consists to estimate a conditional probability distribution $p_{\boldsymbol{\theta}}(y|\boldsymbol{x}, \boldsymbol{C})$, where $\boldsymbol{C}$ is the neighborhood of the pixel $\boldsymbol{x}$.



(a) Pixel classification



(b) Patch segmentation

Figure 2.7: Illustration of the two approaches for hyperspectral semantic segmentation

Patch segmentation. In order to put more emphasize on the spatial context, patch segmentation consists to simultaneously predict the labels of all pixels in a patch. Therefore,

the training data set $\mathcal{D}_{train}$ is composed of image patches $\boldsymbol{x}^{(i)}$ of size $w \times h \times B$ and partially labeled maps $\boldsymbol{y}^{(i)}$ of size $w \times h$. Compared to pixel classification, the loss function is slightly modified:

$$\begin{aligned}\mathcal{L}^{seg}(\boldsymbol{\theta}; \mathcal{D}_{train}) &= \frac{1}{N}\sum_{i=1}^{N} l^{seg}(\boldsymbol{\theta}; p_{\boldsymbol{\theta}}(\boldsymbol{y}^{(i)}|\boldsymbol{x}^{(i)}), \hat{p}_{data}(\boldsymbol{y}^{(i)}, |\boldsymbol{x}^{(i)})) \\ &= \frac{1}{N}\sum_{i=1}^{N}\frac{1}{M}\sum_{k=1}^{w}\sum_{l=1}^{h} \mathbb{1}_{\boldsymbol{y}_{kl}^{(i)} \neq 0} \cdot l^{c}(\boldsymbol{\theta}; p_{\boldsymbol{\theta}}(\boldsymbol{y}_{kl}^{(i)}|\boldsymbol{x}_{kl}^{(i)}), \hat{p}_{data}(\boldsymbol{y}_{kl}^{(i)}, |\boldsymbol{x}_{kl}^{(i)}))\end{aligned} \tag{2.17}$$

where $\boldsymbol{x}_{kl}^{(i)}$ (respectively $\boldsymbol{y}_{kl}^{(i)}$) is the spectrum (respectively the label) at coordinates (k, l) of the patch $\boldsymbol{x}^{(i)}$ (respectively the map $\boldsymbol{y}^{(i)}$), $\mathbb{1}_{\boldsymbol{y}_{kl}^{(i)} \neq 0}$ equals 1 if $\boldsymbol{y}_{kl}^{(i)} \neq 0$, 0 otherwise, and $M = \sum_{k=1}^{w}\sum_{l=1}^{h} \mathbb{1}_{\boldsymbol{y}_{kl}^{(i)} \neq 0}$.

Next, we briefly present two discriminative models: the support vector machines [Cortes and Vapnik, 1995] that we will use in our experiments and neural networks [Rosenblatt, 1958] that are at the core of our research.

Support Vector Machines (SVMs) were introduced by [Cortes and Vapnik, 1995] for binary classification problems, though it can been generalized to multi-class problems [Weston and Watkins, 1998]. [Sollich, 1999] introduced a probabilistic view of the SVM, defining the likelihood of the model through a mapping π parameterized by $\boldsymbol{\theta} = [\boldsymbol{w}\ b]$:

$$\forall i \in \{-1, 1\},\ \pi_i(\boldsymbol{x}; \boldsymbol{\theta}) = \frac{1}{1 + e^{-2\lambda}} \exp\ \big[-\lambda \cdot \max(0, 1 - y_i(\boldsymbol{w}^T\boldsymbol{x} - b)\big] \tag{2.18}$$

The parameters $\boldsymbol{w}$ and b define a hyperplane that divides the data based on their semantic classes, with a penalty coefficient λ that allows outliers to be misclassified.

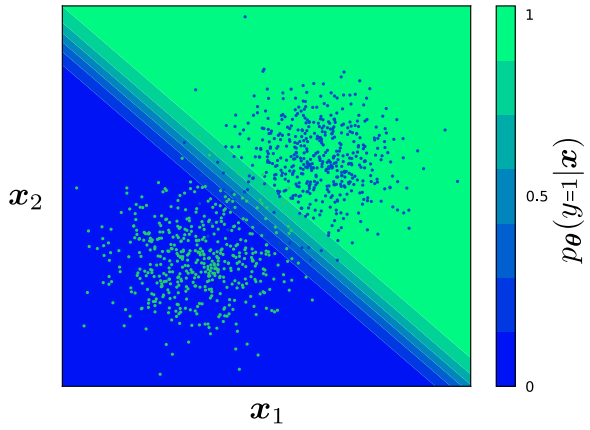


Figure 2.8: Illustration of a 2-class classification problem solved by a linear neural network with one hidden layer or equivalently by a SVM. The class boundary, *i.e.* the hyperplan of the SVM, is defined by the equation $p_{\boldsymbol{\theta}}(y = 1|x) = 0.5$.

Neural networks. The first neural network, the perceptron, was introduced by [Rosenblatt, 1958] and was enhanced much years later as a multi-layer perceptron (MLP) by [Rumelhart et al., 1985], also called a feedforward neural network. Today, MLP are still extensively used in machine learning in order to estimate densities. Formally, a MLP is a combination of affine transformations and nonlinearities:

$$\pi(\boldsymbol{x}; \boldsymbol{\theta}) = \pi^h \circ \ldots \circ \pi^1(\boldsymbol{x}; \boldsymbol{\theta}) \tag{2.19}$$

where h is the number of hidden layers:

$$\forall k \in \{1, \ldots, h\}, \pi^k(\boldsymbol{x}; \boldsymbol{\theta}) = \sigma_k(\boldsymbol{w_k}^T\boldsymbol{x} + b_k) \tag{2.20}$$

where $\boldsymbol{w}_k \in \mathbb{R}^{|\boldsymbol{w}_k|}$ are weights, $b_k \in \mathbb{R}$ is a bias and σ_k is a non linear function, named activation. The set of parameters is referred to as $\boldsymbol{\theta} = [\boldsymbol{w}_1 \ \dots \ \boldsymbol{w}_h \, b_1 \ \dots \ b_h]$. [Hornik et al., 1989] proved that, for any sufficiently smooth function f, there exist optimal parameters $\boldsymbol{\theta}^*$ such that the MLP approximate f with arbitrary accuracy.

For multi-class problems, σ_h is the Softmax function that turns any arbitrary vector in a valid mass function:

$$Softmax(\boldsymbol{x}) = \Big[\frac{\exp \boldsymbol{x}_i}{\sum_j \exp \boldsymbol{x}_j}\Big]_{i\in\{1,\dots,c\}}. \tag{2.21}$$

For hidden layers, *i.e.* intermediate affine transformations, one common state-of-the-art nonlinear activation is the ReLU function:

$$ReLU(\boldsymbol{x}) = \Big[\max(0, \boldsymbol{x}_i)\Big]_{i\in\{1,\dots,|\mathcal{X}|\}} \tag{2.22}$$

Non-parametric classification algorithms Here we would like to make an important digression about non-parametric classification algorithms that do not rely on discriminative modeling. In particular, we would like to briefly introduce Random Forest and k-nearest neighbors algorithms that are common baselines for hyperspectral classification.

· **Random Forest (RF)** [**Breiman, 2001**] is an ensemble of decision trees that recursively partition the (multivariate) data through dichotomous decisions taken on (univariate) independent attributes. Decision trees are optimized with greedy algorithms, known as top-down inductive algorithms, that iteratively find the attribute upon which to split the data that best minimizes a metric (*e.g.* an impurity based criteria) which reflects the generalization error [Rokach and Maimon, 2005]. The different trees among the forest are learned from different subsets of the feature space, fostering robustness to over-fitting.

· **k-nearest neighbors (KNN)** [**Cover and Hart, 1967**; **Fix and Hodges, 1989**] is a classification algorithm that assigns to data samples the class that is the most represented among its k nearest neighbors. Usually, the contribution of each neighbor is weighted such that close neighbors have more influence on the decision than more distant neighbors. The appeal of KNN is that it does not rely on optimization.

1.3 Generative modeling

Recent advances in deep neural networks have enabled a breakthrough in generative modeling over traditional techniques [Ruthotto and Haber, 2021] such as kernel density estimation [Parzen, 1962] or gaussian mixture models [Reynolds et al., 2009]. State-of-the-art deep generative models have indeed demonstrated impressive results to approximate **high dimensional** and **complex** distributions with neural networks [Ruthotto and Haber, 2021]. In hyperspectral imaging, generative models have demonstrated promising results to model the distribution of individual spectra while modeling hyperspectral images as a whole is still an open question [Audebert et al., 2018b; Zhao et al., 2020].

Let's consider a training data set composed of N data points, where N is potentially very large, that are independently and identically sampled from an unknown distribution $p_{data}(\boldsymbol{x})$. In this section, we focus on latent variable deep generative models that assume data has been generated from latent variables $\boldsymbol{z} \in \mathcal{Z}$ and approximate the conditional distribution $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})$ [Ruthotto and Haber, 2021], also called the likelihood. Latent variables are random variable that are never observed: they can be thought as variables encoding the causes of the observation $\boldsymbol{x}$. Latent variables are assumed to follow a tractable probability distribution

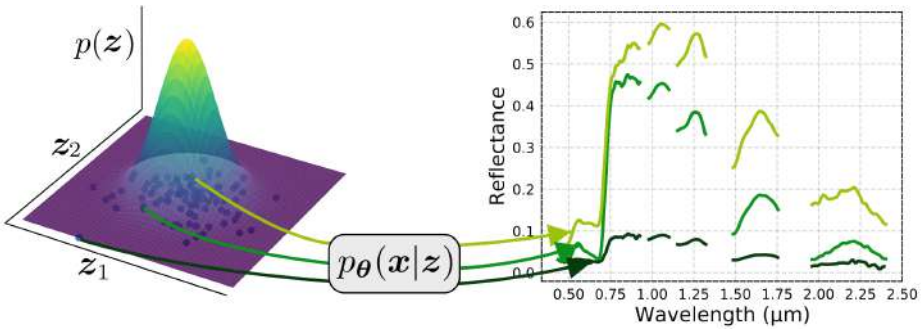


Figure 2.9: Illustration of the generative process of a latent variable deep generative model. First, latent variables $\boldsymbol{z}$ are sampled from a simple distribution, here a multivariate normal. Then, a neural network computes the conditional distribution $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})$, from which data samples are generated.

$p(\boldsymbol{z})$, *i.e.* a simple distribution, such as a multivariate normal, from which we can sample realizations and for which we can compute the density. From this simple distribution, a highly nonlinear neural network $g_{\boldsymbol{\theta}}$ parameterizes the conditional distribution, defined as a multivariate normal:

$$p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}) = \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) \tag{2.23}$$

with the following alternatives:

$$\begin{cases} \boldsymbol{\mu} = g_{\boldsymbol{\theta}}(\boldsymbol{z}), \boldsymbol{\Sigma} = \text{diag}(\boldsymbol{\sigma}) \text{ with } \boldsymbol{\sigma} = cste & (1) \\ [\boldsymbol{\mu}\ \boldsymbol{\sigma}] = g_{\boldsymbol{\theta}}(\boldsymbol{z}), \boldsymbol{\Sigma} = \text{diag}(\boldsymbol{\sigma}) & (2) \\ [\boldsymbol{\mu}\ \boldsymbol{\Sigma}] = g_{\boldsymbol{\theta}}(\boldsymbol{z}) & (3) \end{cases}$$

(1), (2) and (3) are different modeling choices for the likelihood, though (1) is by far the most common, where $\boldsymbol{\Sigma} > 0$ is a hyperparameter that controls how narrow is the likelihood around samples [Ruthotto and Haber, 2021]. Then, the density for a specific data point $\boldsymbol{x}$ can be computed as follows:

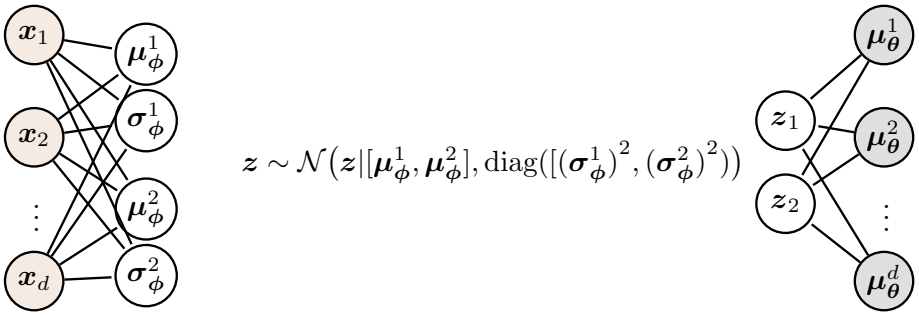
$$p_{\boldsymbol{\theta}}(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{z}\sim p(\boldsymbol{z})} p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}) = \int_{\mathcal{Z}} p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}) p(\boldsymbol{z}) d\boldsymbol{z} \tag{2.24}$$

An important observation is that the density $p_{\boldsymbol{\theta}}(\boldsymbol{x})$ is not a multivariate normal, although the prior and the likelihood are. The density, however, cannot be computed analytically even for a simple conditional distribution parameterized by a shallow nonlinear neural network $g_{\boldsymbol{\theta}}$ [Kingma and Welling, 2014]. Numerically, we could approximate the density with Monte-Carlo sampling:

$$p_{\boldsymbol{\theta}}(\boldsymbol{x}) \approx \frac{1}{K} \sum_{k=1}^{K} p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}^{(k)}) \ ; \ \boldsymbol{z}^{(k)} \sim p(\boldsymbol{z}) \tag{2.25}$$

The problem is that the likelihood $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})$ for a data point $\boldsymbol{x}$ will be near zero for most $\boldsymbol{z}$ [Doersch, 2016], which turns Monte-Carlo sampling impractical.

Therefore, minimizing the KL divergence between the empirical and the approximated data distributions, or equivalently, maximizing the data marginal likelihood is not possible for complex and high dimensional distributions. Next, we present how the variational autoencoder (VAE) [Kingma and Welling, 2014], a state-of-the-art deep generative model, overcomes this limitation. We focus here on the VAE because one major contribution of this thesis is based on it.



Prior $p(\boldsymbol{z}) = \mathcal{N}(\boldsymbol{z}|\mathbf{0}, \boldsymbol{I})$ Posterior $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}) = \mathcal{N}(\boldsymbol{z}|\boldsymbol{\mu}_{\boldsymbol{\phi}}, \text{diag}(\boldsymbol{\sigma}_{\boldsymbol{\phi}}^2))$ Likelihood $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}) = \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_{\boldsymbol{\theta}}, \boldsymbol{\Sigma}_{\boldsymbol{\theta}})$

Figure 2.10: Illustration of a VAE with two-dimension latent variables and a fixed $\boldsymbol{\Sigma}_{\boldsymbol{\theta}}$

1.3.1 Variational Autoencoder [Kingma and Welling, 2014]

A variational autoencoder (VAE) is a generative model, introduced by [Kingma and Welling, 2014], that has raised considerable attention in generative modeling. In order to approximate the density $p_{\boldsymbol{\theta}}(\boldsymbol{x})$, the key idea of the VAE is to sample latent variables $\boldsymbol{z}$ that are likely to have produced the data $\boldsymbol{x}$ [Doersch, 2016]. To this end, [Kingma and Welling, 2014] approximate the posterior distribution $p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x})$ by a parameterized distribution $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$ defined as follows:

$$\begin{cases} q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}) & = \mathcal{N}(\boldsymbol{z}|\mu_z, \text{diag}(\boldsymbol{\sigma}_z)) \\ [\mu_z \; \boldsymbol{\sigma}_z] & = f_{\boldsymbol{\phi}}(\boldsymbol{x}) \end{cases}$$

where $f_{\boldsymbol{\phi}} : \mathcal{X} \longrightarrow \mathbb{R}^{|\mathcal{Z}|} \times \mathbb{R}_+^{|\mathcal{Z}|}$ is a neural network that computes the mean and the standard deviation of the approximated posterior distribution. Modeling the latent variables with an isotropic normal, *i.e.* a multivariate normal parameterized by a diagonal covariance, enforces the independence of the latent variables ($\boldsymbol{z}_i$ and $\boldsymbol{z}_j$ should cause independent variations of the data $\boldsymbol{x}$, for all $i \neq j$).

Thanks to the approximation of the posterior distribution, [Kingma and Welling, 2014] rewrite the data log marginal likelihood, that we would like to maximize (instead of the marginal likelihood for convenience), as follows:

$$\log p_{\boldsymbol{\theta}}(\boldsymbol{x}) = l(\boldsymbol{x}; \boldsymbol{\theta}, \boldsymbol{\phi}) + \underbrace{D_{KL}[q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})||p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x})]}_{\geq 0} \tag{2.26}$$

The $D_{KL}[q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})||p_{\boldsymbol{\theta}}(\boldsymbol{z}|\boldsymbol{x})]$ term denotes the Kullback-Leibler divergence between the approximated posterior and the true posterior for a data point $\boldsymbol{x}$. As far as the true posterior is unknown, the divergence is unknown as well. Since the divergence is positive though, $l(\boldsymbol{x}; \boldsymbol{\theta}, \boldsymbol{\phi})$ is a lower bound of the log marginal likelihood [Kingma and Welling, 2014]:

$$\log p_{\boldsymbol{\theta}}(\boldsymbol{x}) \geq l(\boldsymbol{x}; \boldsymbol{\theta}, \boldsymbol{\phi}) = \underbrace{\mathbb{E}_{z \sim q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})]}_{\text{likelihood term}} - \underbrace{D_{KL}[q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})||p(\boldsymbol{z})]}_{\text{regularization term}} \tag{2.27}$$

The likelihood term measures how the data fits the model, it can be approximated with K samples $\boldsymbol{z}^{(k)}$ from $q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$. The regularization term measures how close is the approximated

posterior from the prior and can be computed analytically in the case of gaussian prior and posterior.

Fig. 2.10 illustrates how the approximated posterior parameterized by $f_{\boldsymbol{\phi}}$, and the likelihood, parameterized by $g_{\boldsymbol{\theta}}$, form an autoencoder framework. The posterior distribution on the latent variables *encode* the data while the likelihood *decode* latent variables.

Until now, we have not mentioned how the VAE can be used to predict semantic classes from data. [Kingma et al., 2014] introduced a simple extension of the traditional VAE that consists in adding a class variable y in addition to previous continuous latent variables $\boldsymbol{z}$:

$$\begin{cases} p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z},y) & = \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_{\boldsymbol{\theta}},\boldsymbol{\Sigma}_{\boldsymbol{\theta}}) \\ [\boldsymbol{\mu}_{\boldsymbol{\theta}}\ \boldsymbol{\Sigma}_{\boldsymbol{\theta}}] & = g_{\boldsymbol{\theta}}(\boldsymbol{z},y) \end{cases} \quad ; \quad \begin{cases} q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x},y) & = \mathcal{N}(\boldsymbol{z}|\boldsymbol{\mu}_{\boldsymbol{\phi}},\boldsymbol{\Sigma}_{\boldsymbol{\phi}}) \\ q_{\boldsymbol{\phi}}(y|\boldsymbol{x}) & = \mathrm{Cat}(y|\pi(\boldsymbol{x};\boldsymbol{\phi})) \\ [\boldsymbol{\mu}_{\boldsymbol{\phi}}\ \boldsymbol{\Sigma}_{\boldsymbol{\phi}}] & = f_{\boldsymbol{\phi}}(\boldsymbol{x},y) \end{cases} \tag{2.28}$$

where $\pi : \mathcal{X} \longrightarrow [0,1]^C$ is an additional neural network that estimates the categorical conditional distribution of y given $\boldsymbol{x}$. Considering a data set composed of N labeled points $\{(\boldsymbol{x}^{(1)},y^{(1)}),\dots,(\boldsymbol{x}^{(N)},y^{(N)})\}$ and M unlabeled points $\{\boldsymbol{x}^{(N+1)},\dots,\boldsymbol{x}^{(N+M)}\}$, [Kingma et al., 2014] compute a lower bound $\mathcal{L}(\boldsymbol{x},y;\boldsymbol{\theta},\boldsymbol{\phi})$ of the log likelihood $p_{\boldsymbol{\theta}}(\boldsymbol{x},y)$ of labeled data and a lower bound $\mathcal{U}(x;\boldsymbol{\theta},\boldsymbol{\phi})$ of the log marginal likelihood $p_{\boldsymbol{\theta}}(\boldsymbol{x})$ of unlabeled data, defined as follows, respectively:

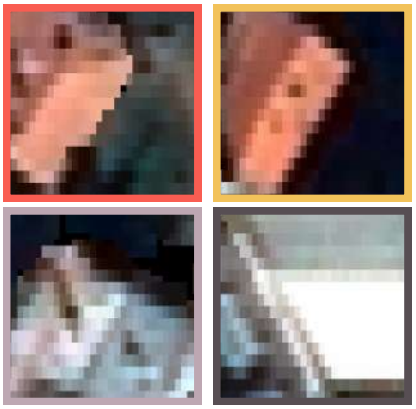
$$-\mathcal{L}(\boldsymbol{x},y;\boldsymbol{\theta},\boldsymbol{\phi}) = \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x},y)}[\log\, p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z},y) + \log\, p(y)] - D_{KL}(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x},y)||p(\boldsymbol{z})) \leq \log p_{\boldsymbol{\theta}}(\boldsymbol{x},y) \tag{2.29}$$

$$-\mathcal{U}(\boldsymbol{x};\boldsymbol{\theta},\boldsymbol{\phi}) = -\sum_{y}\left[q_{\boldsymbol{\phi}}(y|\boldsymbol{x})\mathcal{L}(\boldsymbol{x},y;\boldsymbol{\theta},\boldsymbol{\phi})\right] + H[q_{\boldsymbol{\phi}}(y|\boldsymbol{x})] \leq \log p_{\boldsymbol{\theta}}(\boldsymbol{x}) \tag{2.30}$$

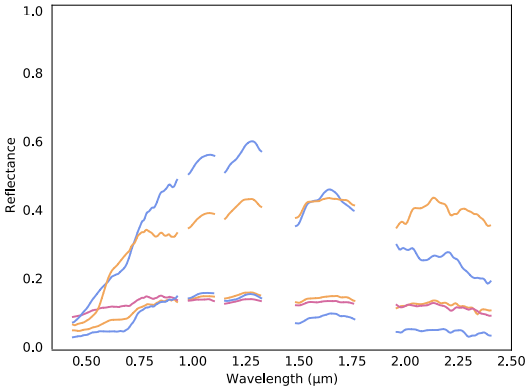
> Because samples from a complex and high-dimensional data distribution can be very different though having high semantic similarities, learning densities is difficult. Probabilistic latent models such as VAEs circumvent this issue by learning representations of the data that lie in a low-dimensional space. This strategy is not confined to generative models. Usually, discriminative models also learn a deterministic mapping $f_{\boldsymbol{\phi}} : \mathcal{X} \longrightarrow \mathcal{Z}$ and estimate the conditional distribution $p_{\boldsymbol{\theta}}(y|\boldsymbol{z})$ over the representation rather than the raw data itself. In the next section, we present the state-of-the-art techniques to learn representations.

2 Representation learning

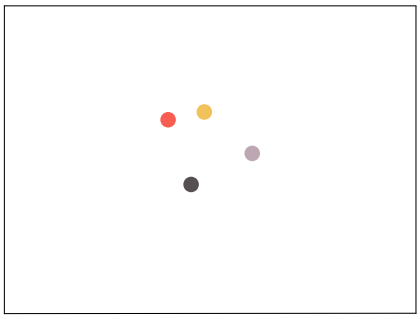
Estimating probability distributions from high-dimensional data is usually much more difficult than for low-dimensional data. For the purpose of explanation, consider that we would like to approximate the density $p(\boldsymbol{x})$ of satellite images or of hyperspectral data. Fig. 2.11a illustrates that different images with similar contents can be far from each other in the data space: the euclidean distances between the top left image and the top right, bottom left and bottom right images are, respectively, equal to 11.8, 19.8 and 27.5. In the same manner, spectra in Fig. 2.11b from the same semantic classes are far from each other in the input space. The goal of representation learning is to optimize a mapping $f_\phi : \mathcal{X} \longrightarrow \mathcal{Z}$ where $|\mathcal{Z}| \ll |\mathcal{X}|$ such that $f_\phi(\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)})$ is a good representation (or features, view, summary...) of the data $\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\}$. Intuitively, we expect samples that share semantic properties to have similar representations, as illustrated in Fig. 2.11.



(a) High resolution satellite images acquired over Toulouse, France



(b) Hyperspectral spectra of dry vegetation, tile and asphalt.



(c) Illustration of some ideal 2D representation of the images

(d) Illustration of some ideal 2D representation of the spectra

Figure 2.11: Illustration of intra-class variability of various data, that makes density estimation without representation learning, extremely challenging.

Another way of looking at representation learning is to consider it as a way to generalize. Indeed, an over-parameterized MLP could reach a 100% training accuracy for an image classification task. However, as far as a MLP is sensitive to object translations and object rotations, its accuracy would probably drop for new images. In contrast, representing images with features that are invariant to object translations and object rotations would probably lead to a more robust classifier: the choice of the representation is an inductive bias.

We present the state-of-the-art techniques to extract spatial features from images in section 2.1, spectral features from hyperspectral data in section 2.2 and spatial-spectral features in section 2.3.

2.1 Spatial features

0	0	0
0	1	0
0	0	0

(a) Original image (identity kernel)

$\frac{1}{16}$	$\frac{1}{8}$	$\frac{1}{16}$
$\frac{1}{8}$	$\frac{1}{4}$	$\frac{1}{8}$
$\frac{1}{16}$	$\frac{1}{8}$	$\frac{1}{16}$

(b) Smoothed image (gaussian kernel)

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

(c) Horizontal contours (Sobel filter [Sobel, 2014])

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

(d) Vertical contours (Sobel filter [Sobel, 2014])

Figure 2.12: Examples of convolutions. **Top**: convolved images. **Bottom**: convolution kernels.

Convolutional Neural Networks (CNNs). CNNs have become the state of the art to extract discriminative spatial features for image classification and image segmentation, both for natural images [Hao et al., 2020; Krizhevsky et al., 2012; Touvron et al., 2019] and for remote sensing images [Audebert et al., 2017, 2019; Castillo-Navarro et al., 2021]. [Lecun et al., 1998] first introduced discrete convolutions in neural networks for image classification though they were already commonly used in signal processing [Damelin and Miller, 2011], for instance by the SIFT [Lowe, 1999] and HOG [Dalal and Triggs, 2005] descriptors. The convolution of an image I of size $W \times H$ by a kernel K of size $k_W \times k_H$ is defined as follows:

$$\forall (m,n) \in [1+p, W-p] \times [1+q, H-q],\ K * I[m,n] = \sum_{i=-p}^{p} \sum_{j=-q}^{q} I[m-i, n-j] \cdot K[i,j] \tag{2.31}$$

where $p = \frac{k_W - 1}{2}$ and $q = \frac{k_H - 1}{2}$. Examples of convolutions are shown in Fig. 2.12: very simple kernels such as the Sobel filters [Sobel, 2014] can be very informative of the geometry of the scene by extracting contours. Note that a convolved image has smaller dimensions than the input image, precisely $2p$ and $2q$ less pixels on the width and height, respectively, as far as the convolutions cannot be applied on the edges of the image. To keep the original dimensions, padding consists in adding $2p$ and $2q$ additional pixels around the image with arbitrary values (usually zeros).

In CNNs, the parameters of convolution kernels are learned. In Fig. 2.13, the kernels of the first convolutional layer of the deep network AlexNet [Krizhevsky et al., 2012] are shown. They were optimized for a classification task on the ImageNet data set [Deng et al., 2009] which consists in 224×224 natural images (images of *cats*, *dogs*, *planes*...). The kernels resemble Gabor filters [Gabor, 1946; Granlund, 1978] which are Gaussian kernels modulated by a sinusoidal wave. They allow to extract low-abstraction features such as edges that are agnostic to the end task. In contrast, [Goodfellow et al., 2014] claim that deeper convolutional layers learn kernels that extract more abstract features which are more specialized on the task at hand. In order to enforce the invariance of the representation to small translations, convolutional layers are usually followed by pooling operations which consist in subsampling the convolved images. Pooling operations also remove redundancy and decrease the size of the data, hence the size of the model and the computational requirements. Another

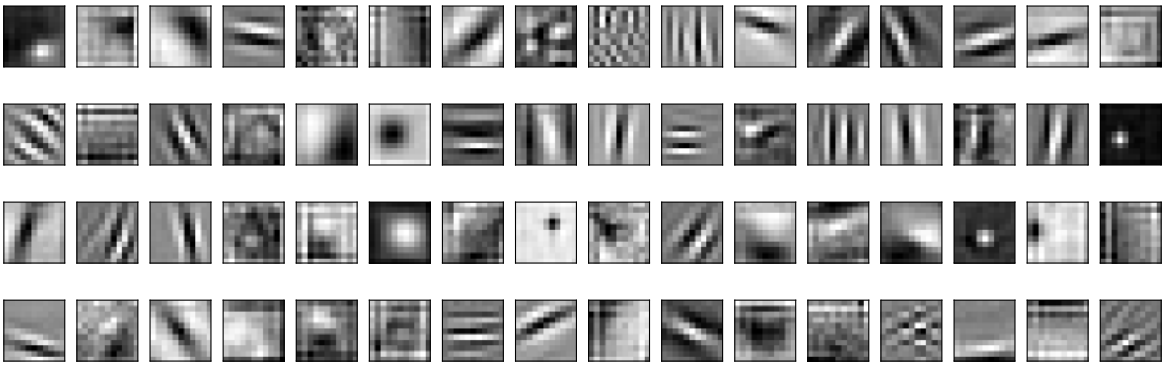


Figure 2.13: Kernel convolutions of the first layer of AlexNet [Krizhevsky et al., 2012] learned from a classification task on ImageNet [Deng et al., 2009]

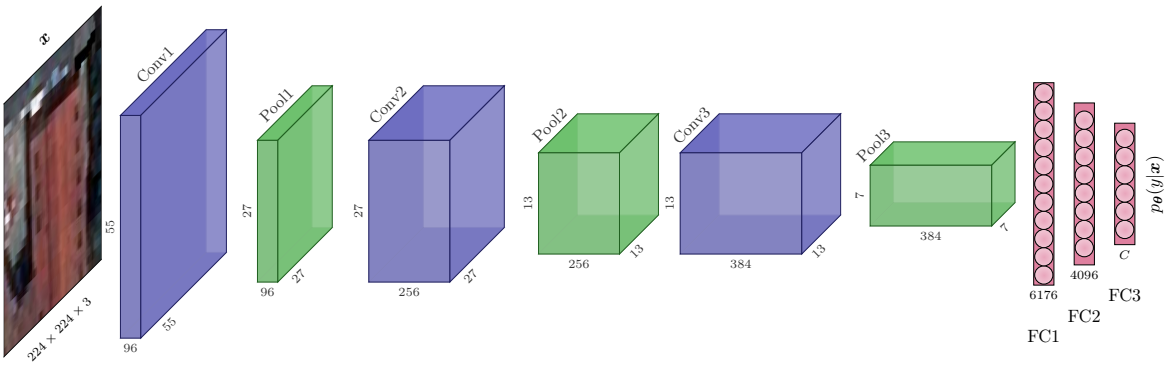


Figure 2.14: Illustration of an AlexNet-like CNN that performs classification over c classes (figure freely adapted from [Audebert, 2018]). The first convolutional layer applies 3×96 kernels of size 11×11 with a stride of 4 on a RGB image, resulting in a $55 \times 55 \times 96$ activation map. A 2×2 pooling operation reduces by 2 the spatial dimensions of the activation map. Two layers of 5×5 convolutions with a padding of 2 pixels and pooling operations follow. The resulted features are flattened in a $7 \times 7 \times 384$ vector, fed in the dense layers that computes the conditional approximation $p_{\boldsymbol{\theta}}(y|x)$.

way to limit redundancy is to increase the striding of the convolution, *i.e.* the number of pixels through which the kernel moves after each operation. Finally, CNNs complement convolutions with dense layers to perform classification. In philosophy, although not totally true in practice, dense layers of CNNs estimate the conditional distribution $p_{\boldsymbol{\theta}}(y|\boldsymbol{x})$ through linear transformations of the representations (or features) that are nonlinear transformations of the input data, computed with the convolutional layers followed by nonlinear activations. An illustration of an AlexNet-like CNN is shown in Fig. 2.14.

A major advantage of convolutional layers over dense layers lies in their **weight sharing** property: the same convolution kernel applies at different places on the image. A dense layer could compute the same transformation, yet at the cost of much more parameters [Goodfellow et al., 2016].

Fully convolutional networks (FCN). FCNs are neural networks with convolutional layers only, introduced by [Long et al., 2015] for semantic segmentation. The U-Net [Ronneberger et al., 2015] is a famous FCN with an autoencoder architecture that demonstrated impressive results for semantic segmentation of medical images. The U-net architecture, shown in Fig. 2.15, relies on three fundamental ideas: 1) the spatial context is encoded in the latent space, whose lower dimension enforces small details to be ignored, yielding smooth semantic maps; (2) the decoder, made of transposed convolutions[1], produces a map of conditional

[1] A transposed convolution is the *inverse* of a convolution in the sense that, if it takes as input a convolved

distributions $p_{\boldsymbol{\theta}}(\boldsymbol{y}|\boldsymbol{x}) = \left[p_{\boldsymbol{\theta}}(y_{ij}|\boldsymbol{x})\right]_{(i,j)\in[1,W]\times[1,H]}$ from the latent space; (3) input data and intermediate convolved data are concatenated at different levels of the decoder in order to preserve the geometry and small details of the scene.

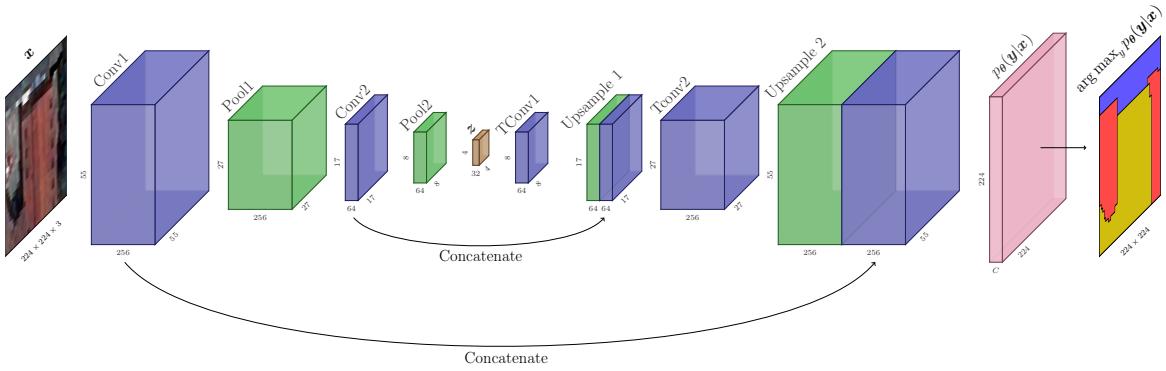


Figure 2.15: Illustration of a U-Net-like FCN that performs semantic segmentation over c classes. The first convolutional layer applies 3×256 kernels of size 11×11 with a stride of 4 on a RGB image, resulting in a $55 \times 55 \times 256$ activation map. A 2×2 pooling operation reduces by 2 the spatial dimensions of the activation map. A second convolutional layer applies 256×64 kernels of size 11×11, followed by a 2×2 pooling operation. A third convolutional layer applies 64×32 kernels of size 5×5, which yields a $4 \times 4 \times 32$ representation z of the input image. The decoding path of the network is symmetric to the encoding path: transposed convolutions and upsampling are used to recover the original dimensions of the image. Additionally, the first and second convolved images are concatenated along the spectral dimension during the decoding operations.

The U-Net architecture has been extensively modified and extended. Most state-of-the-art segmentation networks are based on its encoder-decoder architecture, such as SegNet [Badrinarayanan et al., 2017] that was adapted in [Audebert et al., 2018a] for remote sensing applications. U-net like FCNs have demonstrated high capabilities to segment satellite and airborne images with medium to low spectral resolutions [Stoian et al., 2019] and, in particular, to deal with multi-scale features [Audebert et al., 2018a], which is imperative to extract contextual information.

2.2 Spectral features

Spectral indices are linear combination of bands, eventually normalized. They are either designed based on prior knowledge on how chemical properties of matter correlate with spectral features such as absorption peaks or on empirical observations. For instance, a widely used spectral index in the remote sensing community to detect vegetation and characterize its chlorophyll content is the Normalized Difference Vegetation Index (NDVI):

$$NDVI(\boldsymbol{x}) = \frac{\boldsymbol{x}_{0.80} - \boldsymbol{x}_{0.65}}{\boldsymbol{x}_{0.80} + \boldsymbol{x}_{0.65}} \tag{2.32}$$

where $\boldsymbol{x}_\lambda$ denotes the reflectance at the wavelength λ. Other spectral indices commonly used to characterize vegetation, water surfaces and built-up surfaces are (non-exhaustively) listed in Tab. 2.1.

Spectral indices may be used for semantic mapping alongside hand-crafted decision rules, such as:

$$p(y = vegetation|\boldsymbol{x}) = \begin{cases} 1 \text{ if } NDVI(\boldsymbol{x}) > \tau \\ 0 \text{ else} \end{cases} \tag{2.33}$$

image (from an original image I), then it will output an image I' with the same dimensions than I, though I and I' will have different values.

Table 2.1: Examples of commonly used spectral indices. $\boldsymbol{x}_\lambda$ denotes the reflectance at the wavelength λ.

Spectral index	
Vegetated surfaces	
Advanced Normalized Vegetation Index (ANVI) [Peña-Barragán et al., 2007]	$\frac{\boldsymbol{x}_{0.80}-\boldsymbol{x}_{0.45}}{\boldsymbol{x}_{0.80}+\boldsymbol{x}_{0.45}}$
Chlorophyll Index (CI) [Bausch and Khosla, 2010]	$\frac{\boldsymbol{x}_{0.80}}{\boldsymbol{x}_{0.55}} - 1$
Normalized Difference Infrared Index (NDII) [Cheng et al., 2013]	$\frac{\boldsymbol{x}_{0.80}-\boldsymbol{x}_{1.7}}{\boldsymbol{x}_{0.80}+\boldsymbol{x}_{1.7}}$
Soil Adjusted Vegetation Index (SAVI) [Huete, 1988]	$\frac{(\boldsymbol{x}_{0.80}-\boldsymbol{x}_{0.55})(1+l)}{\boldsymbol{x}_{0.80}+\boldsymbol{x}_{0.55}+l}, l > 0$
Water surfaces	
Modified Normalized Difference Water Index (MNDWI) [Xu, 2006]	$\frac{\boldsymbol{x}_{0.55}-\boldsymbol{x}_{2.2}}{\boldsymbol{x}_{0.55}+\boldsymbol{x}_{2.2}}$
Built-up surfaces	
Normalized Difference Built-up Index (NDBI) [Zha et al., 2003]	$\frac{\boldsymbol{x}_{2.2}-\boldsymbol{x}_{0.80}}{\boldsymbol{x}_{2.2}+\boldsymbol{x}_{0.80}}$
Visible green-based built-up index (VGBI) [Estoque and Murayama, 2015]	$\frac{\boldsymbol{x}_{0.55}-\boldsymbol{x}_{0.80}}{\boldsymbol{x}_{0.55}+\boldsymbol{x}_{0.80}}$
Built-up Area Extraction Method (BAEI)	$\frac{\boldsymbol{x}_{0.65}+0.3}{\boldsymbol{x}_{0.55}+\boldsymbol{x}_{1.6}+\epsilon}$
Index-based Built-up Index (IBI) [Xu, 2008]	$\frac{NDBI-(SAVI+MNDWI)/2}{NDBI+(SAVI+MNDWI)/2}$

where τ is a threshold tuned on a validation set, or as inputs to a machine learning model π parameterized by $\boldsymbol{\theta}$:

$$p(y = vegetation|\boldsymbol{x}) = \pi(NDVI(\boldsymbol{x}); \boldsymbol{\theta}) \tag{2.34}$$

Representing hyperspectral data with spectral indices is a strong inductive bias: we assume that much information is contained in small spectral intervals. This assumption is very similar to the parsimony property sometimes enforce to machine learning models, for instance with a regularization of the L1 norm of the model parameters [Santosa and Symes, 1986].

Dimension reduction. In the remote sensing community, dimension reduction is sometimes manually performed based on expert knowledge by selecting a subset of spectral bands. Statistical reduction techniques are more common though, such as the Principal Component Analysis (PCA) [Tipping and Bishop, 1999] that represent data in a lower dimensional space where every dimensions are a linear combination of the original dimensions such that the variance per dimension is maximized. t-distributed stochastic neighbor embedding (t-SNE) [Van der Maaten and Hinton, 2008] is a common alternative to PCA for data visualization. In contrast to PCA, t-SNE is a nonlinear transormation of the data that conserves the distance in the representation space (*i.e.* close data points in the input space are also close in the feature space).

Spectral convolutions. Convolutional layers can be trivially extended to 1D signals. 1D

convolutional layers can also be compared to 1D Gabor filter, as much as they highlight or diminish the contribution of some frequencies to the signal. The main drawback of spectral CNNs is their limited receptive field. The receptive field of a CNN, illustrated in Fig. 2.16, is the region a unit of the network depends on [Luo et al., 2016]. Obvious options to increase the receptive field of a CNN are to increase the kernel size or to increase the depth of the network. Because of their limited receptive field, CNNs cannot learn correlations between very far apart spectral bands (such as bands in the visible and bands in the short-wave infrared) that could be very informative on the semantic class.

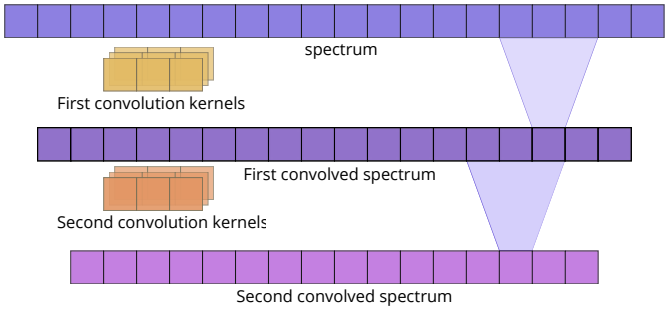


Figure 2.16: Illustration of the receptive field of a spectral CNN. With two consecutive 1×3 spectral convolutions, a channel of the second convolved spectrum is a function of the a 1×5 neighborhood of the input.

2.3 Spectral-spatial features

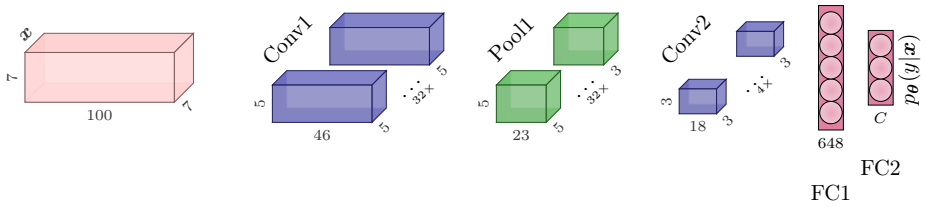


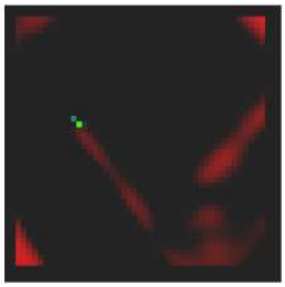
Figure 2.17: Architecture of a spatial-spectral CNN. The input is a $7 \times 7 \times 100$ hyperspectral patch and the output is a distribution over the c classes of the pixel at the center of the input patch. The first convolutional layer applies 1×32 kernels of size $3 \times 3 \times 9$ with a stride of $1 \times 1 \times 2$. Then, a $1 \times 1 \times 2$ pooling operation is applied before a second convolutional layer with 32×4 kernels of size $3 \times 3 \times 11$ with a stride of $1 \times 1 \times 2$. Finally, two dense layers map the features to the conditional distribution.

Spectral-spatial CNNs. Convolutional layers can be extended to three dimensions, handling simultaneously the spectral and the spatial dimensions. They are usually applied to small hyperspectral patches to predict the semantic class of the pixel in the center. They have empirically demonstrated their superiority over 1D or 2D CNNs for hyperspectral image segmentation [Audebert et al., 2019; Li et al., 2017] on various hyperspectral images, though they are prone to deform the geometry of the scene [Derksen et al., 2020] by rounding off the edges. False color composition of activation maps (*i.e.* convolved images) computed with the 3D CNN introduced by [Li et al., 2017] are shown in Fig. 2.18. We see that kernel 1 is sensitive to the presence of tile while kernel 2 seems to highlight vegetation in the shadows as well as trees.

Tab. 2.2, taken from the comparative review of [Audebert et al., 2019], compares the accuracy of a SVM, a MLP, a spectral CNN and a spectral-spatial CNN on segmentation tasks over several public hyperspectral data sets. When the spatial resolution is high enough ($\sim$ 1m),

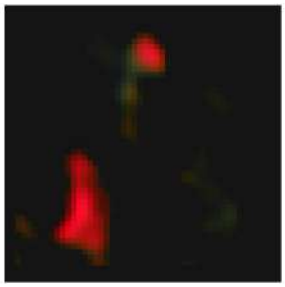
(a) False color composition of the input hyperspectral image

(b) Convolved image by kernel 1

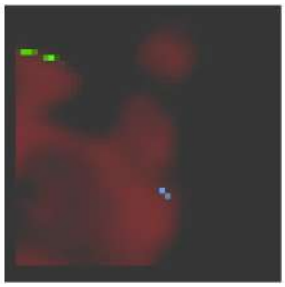


(c) Convolved image by kernel 2

(d) False color composition of the input hyperspectral image



(e) Convolved image by kernel 1



(f) Convolved image by kernel 2

Figure 2.18: Examples of activation maps computed by spatial-spectral convolutions on hyperspectral images

Table 2.2: Image segmentation accuracy on several hyperspectral data sets [Audebert et al., 2019]

Method	Indian Pines	Pavia University	Houston University
SVM	81%	70%	43%
MLP	**83**%	77%	41%
1D CNN [Hu et al., 2015]	**83**%	81%	47%
3D CNN [Li et al., 2017]	75%	**84**%	**49**%

the spatial-spectral CNN outperforms over models in term of accuracy while it performs worse when the spatial resolution is low ($\sim$ 20m).

> If the choice of a specific representation technique, such as convolutions, rather than others should lead to better generalization performances, how the values of the convolution kernels should be chosen? What are the inductive biases that foster some model parameters, *e.g.* convolution kernels, rather than others? In the next section, we present the state-of-the-art optimization techniques to leverage the training labeled samples as well as the unlabeled samples.

3 Optimization algorithms

The discriminative and generative modeling techniques presented in section 1 consist in estimating probability distributions through the optimization of a loss function $\mathcal{L}(\boldsymbol{\theta}; \mathcal{D}_{train})$ with respect to parameters $\boldsymbol{\theta}$, according to a training data set $\mathcal{D}_{train}$:

$$\min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}; \mathcal{D}_{train}) \tag{2.35}$$

In most cases, the optimization problem 2.35 has no closed-form solutions, which motivates the use of iterative algorithms. For density estimation with neural networks, the Stochastic Gradient Descent (SGD) algorithm [Kiefer and Wolfowitz, 1952; Robbins and Monro, 1951], that builds on Gradient Descent (GD) [Cauchy et al., 1847], has become ubiquitous. The principle of GD is to iteratively minimize the objective function $\mathcal{L}(\boldsymbol{\theta}; \mathcal{D}_{train})$ by updating the parameters $\boldsymbol{\theta}$ in the opposite direction of gradients $\nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}; \mathcal{D}_{train})$. When the data is very large, a single computation of the gradient can be computationally expensive. Therefore, the (batch) SGD consists in estimating the gradient from samples of the data:

$$\nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}; \mathcal{D}_{train}) = \nabla_{\boldsymbol{\theta}} \frac{1}{N} \sum_{i=1}^{N} l(\boldsymbol{\theta}; \boldsymbol{x}^{(i)}) \approx \nabla_{\boldsymbol{\theta}} \mathcal{L}^{SGD}(\boldsymbol{\theta}; \mathcal{D}_{train}) = \nabla_{\boldsymbol{\theta}} \frac{1}{B} \sum_{\boldsymbol{x}^{(i)} \in \mathcal{B}} l(\boldsymbol{\theta}; \boldsymbol{x}^{(i)}) \tag{2.36}$$

where $\mathcal{B}$ is a random subset of $\mathcal{D}_{train}$ of size $B \ll N$. Most of the time, deep learning models are over-parameterized, *i.e.* the size of $N \ll |\boldsymbol{\theta}|$. In this case, the optimization problem 2.35 admits several global minima. Obviously, different global solutions $\boldsymbol{\theta}^1$ and $\boldsymbol{\theta}^2$ have the same global minima with respect to the training data set. However, the objective would likely take different values on unseen data $\mathcal{D}_{test}$ [Wu et al., 2018]:

$$\mathcal{L}(\boldsymbol{\theta}^1; \mathcal{D}_{test}) \neq \mathcal{L}(\boldsymbol{\theta}^2; \mathcal{D}_{test}) \tag{2.37}$$

To *choose* a global minima rather than another, machine learning models need inductive biases, *i.e.* properties of the model that are assumed to improve generalization.

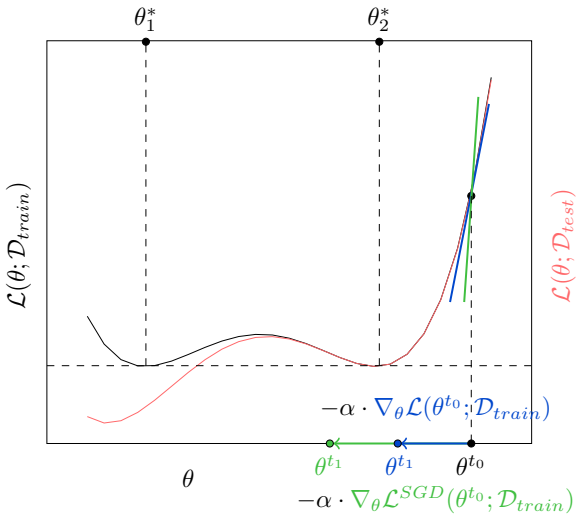


Figure 2.19: Illustration of GD and SGD for a 1D dimensional model parameterized by θ and optimized over a training data set $\mathcal{D}_{train}$.

First, we review common inductive biases for supervised algorithms in section 3.1. Then, we review state-of-the-art semi-supervised techniques in section 3.2.

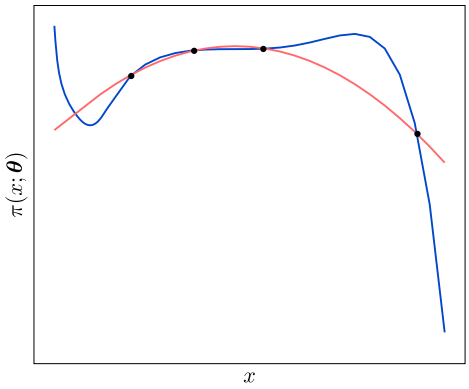


Figure 2.20: Example of two regression models that have reached zero loss on the training data (illustrated as black dots). A common assumption is that the simpler model will generalize better than the more complex model.

3.1 Supervised learning

Regularization. A very common inductive bias is the assumption whereby simpler model generalize better. This idea is illustrated in Fig. 2.21 and is related to the smoothness assumption that states that close samples in the input space should have close predictions. One measure of model simplicity is the average L2 norm of model parameters. Adding a penalization of the L2 norm to the main objective function was first introduced for regression problems by [Hilt and Seegrist, 1977] as Ridge regression. It is easily applied to classification tasks and is referred as L2 regularization, or weight decay:

$$\mathcal{J}(\boldsymbol{\theta};\mathcal{D}_{train}) = \mathcal{L}(\boldsymbol{\theta};\mathcal{D}_{train}) + \lambda||\boldsymbol{\theta}||_2^2 \tag{2.38}$$

where λ is a penalty coefficient that weights the influence of the regularization term. It prevents the model to put too much emphasize on a specific feature, or to a specific combination of features. In the other hand, L1 regularization [Santosa and Symes, 1986], briefly mentioned in section 2, fosters model parsimony:

$$\mathcal{J}(\boldsymbol{\theta};\mathcal{D}_{train}) = \mathcal{L}(\boldsymbol{\theta};\mathcal{D}_{train}) + \lambda||\boldsymbol{\theta}||_1 \tag{2.39}$$

where $||\boldsymbol{\theta}||_1 = \sum_i |\boldsymbol{\theta}_i|$.

Another very popular regularization strategy for MLPs is dropout [Srivastava et al., 2014]: it consists to randomly zero out weights of the network at each forward pass (see the illustration in Fig. 2.22). The implicit inductive bias used by dropout is that weights within a hidden layer of a MLP should not co-adapt too much during the training, which is close to weight decay.

Data augmentation. When representation techniques fail to be robust to slight geometric or radiometric changes, data augmentation is a widely used technique to improve the robustness of machine learning models for supervised tasks. Data augmentation consists in increasing the training data set by applying random transformations to the existing data, such as image rotations, image translations [Wang et al., 2017], random noise addition [Sietsma and Dow, 1991]... More recent approaches use generative models to simulate new data [Wang et al., 2017], [Audebert et al., 2018b] for hyperspectral imaging.

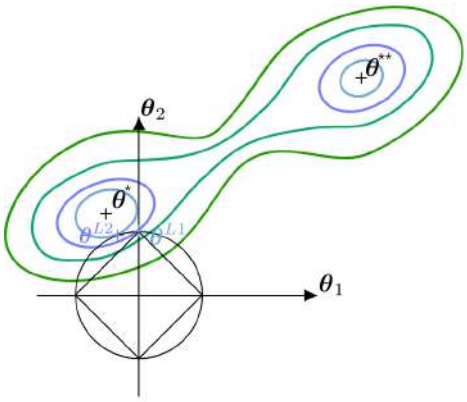


Figure 2.21: Illustration of L1 and L2 regularization on a model parameterized by two-dimensional parameters $\boldsymbol{\theta}$. The contours of the objective function are plotted from green (highest values) to blue (lowest values). Black contours are specified by the equations $||\boldsymbol{\theta}||_2^2 = 1$ and $||\boldsymbol{\theta}||_1 = 1$. $\boldsymbol{\theta}^*$ and $\boldsymbol{\theta}^{**}$ denote two possible optimal parameters. Minimizing a linear combination of the objective function and the regularization term fosters model parameters that are close to an optimal solution while minimizing the parameters norm. We denote such solutions as $\boldsymbol{\theta}^{L1}$ and $\boldsymbol{\theta}^{L2}$ for the regularization on the L1 norm and L2 norm, respectively.

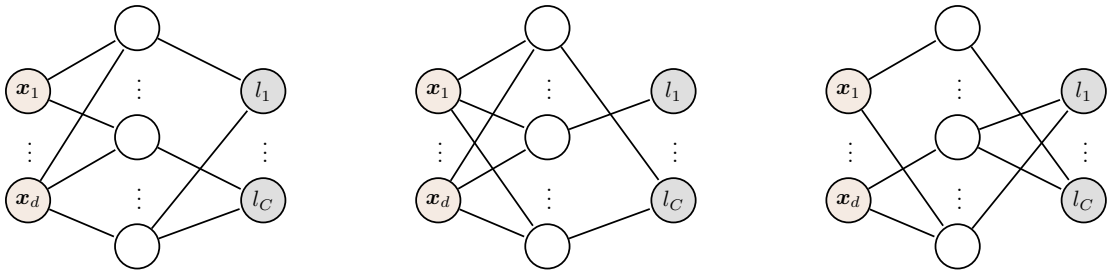


Figure 2.22: Different network configurations with a 0.5 probability of dropout

3.2 Semi-supervised learning

Semi-supervised algorithms aim to leverage the structure of unlabeled data. Generative models, by nature, are optimized with unsupervised techniques. As we have seen in section 1, some generative models can be enhanced to benefit from the information of few labeled samples also, such as the semi-supervised VAE [Kingma et al., 2014]. Other state-of-the-art generative models such as Generative Antagonist Nets (GANs) [Goodfellow et al., 2014] or normalizing flows [Rezende and Mohamed, 2015] have also been extended to semi-supervised settings, for instance by [Spurr et al., 2017] that introduced a semi-supervised version of InfoGAN [Chen et al., 2016] and [Izmailov et al., 2020] that developed a semi-supervised algorithm for normalizing flows. [Van Engelen and Hoos, 2020] provide an exhaustive survey on transductive and inductive semi-supervised methods. Transductive methods optimize over the predictions themselves while inductive methods optimize over predictive models. Inductive methods include pseudo-labeling, or self-labeled, approaches [Triguero et al., 2015] (where the labeled data set is iteratively enlarged by the predictions of the model), unsupervised preprocessing (such as feature extraction with autoencoders [Salah et al., 2011] or pretraining [Erhan et al., 2010]) and regularization techniques that mainly rely on two inductive biases: the manifold assumption and the smoothness assumption.

Manifold regularization. The inductive bias manifold regularization [Belkin et al., 2006] introduces in learning is that samples from the same semantic class lie in low-dimensional subset, called a manifold [Van Engelen and Hoos, 2020]. Combined with the smoothness

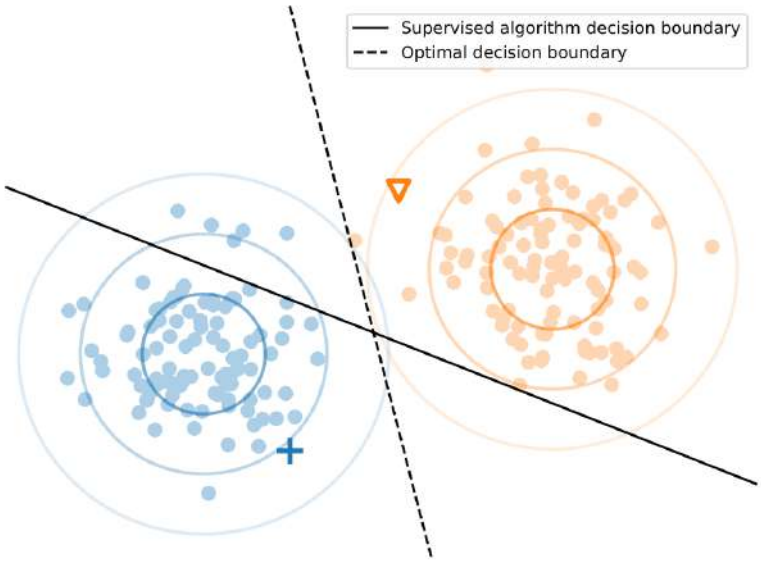


Figure 2.23: A basic example of binary classification in the presence of unlabeled data. The unlabeled data points are colored according to their true label. The colored, unfilled circles depict the contour curves of the input data distribution corresponding to standard deviations of 1, 2 and 3. The cross and the triangle are labeled samples. Figure and caption from [Van Engelen and Hoos, 2020]

assumption, it has opened the path to a myriad of unsupervised techniques to regularize the parameters of neural networks, such as additional reconstruction losses in the latent space [Ranzato and Szummer, 2008; Rifai et al., 2011; Weston et al., 2012], perturbation-based approaches with virtual adversarial training [Miyato et al., 2018], temporal ensembling [Laine and Aila, 2016] that uses a consistency loss on different stochastic model outputs from the same input, or additional unsupervised tasks such as relaxed k-means or reconstruction losses in the context of semantic segmentation [Castillo-Navarro et al., 2021]. The principle of an additional reconstruction task is illustrated in Fig. 2.24. A neural network π_0 encodes the data in a low-dimensional latent space. From the latent variables $\boldsymbol{z}$, one neural network π_1 computes the conditional distribution $p_{\boldsymbol{\theta}}(\boldsymbol{y}|\boldsymbol{z})$ and another neural network π_2 computes the reconstruction of the input.

> State-of-the-art optimization techniques make some assumptions, namely inductive biases, on the properties a machine learning model should admit to generalize on new data. As far as labeled data is usually scarce and that unlabeled data is usually large, semi-supervised techniques have raised considerable attention in the past years.

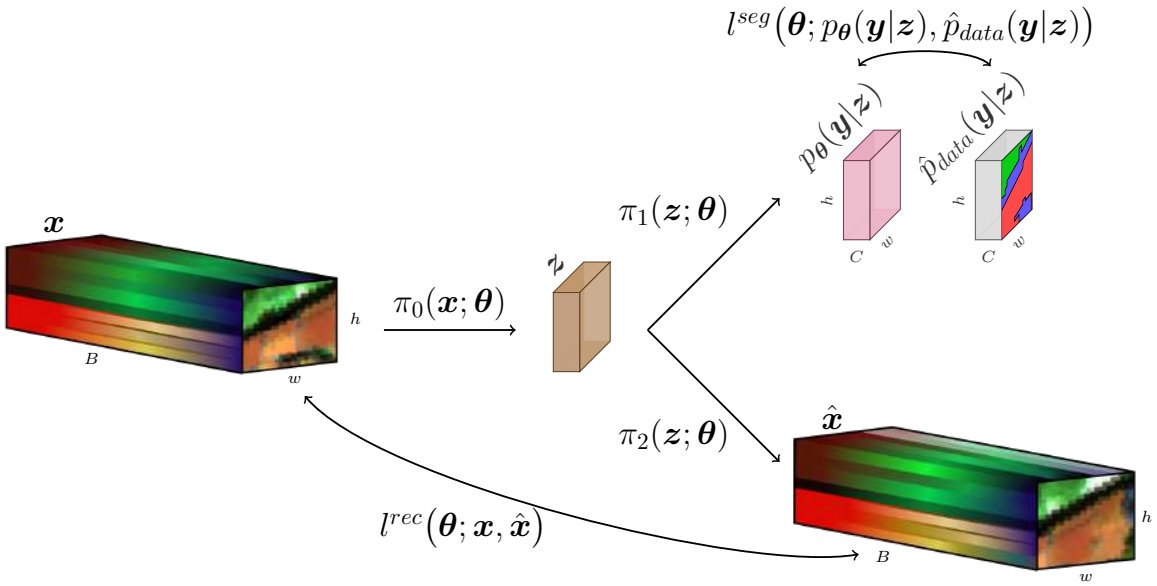


Figure 2.24: Illustration of a semi-supervised learning process with a segmentation task and a reconstruction task, as introduced in [Castillo-Navarro et al., 2021] for segmentation of high spatial resolution remote sensing images.

4 References

Audebert, N. (2018). *Classification de données massives de télédétection*. PhD thesis, Université Bretagne Sud. 42

Audebert, N., Le Saux, B., and Lefèvre, S. (2017). Semantic segmentation of earth observation data using multimodal and multi-scale deep networks. In *Computer Vision–ACCV 2016: 13th Asian Conference on Computer Vision, Taipei, Taiwan, November 20-24, 2016, Revised Selected Papers, Part I 13*, pages 180–196. Springer. 41

Audebert, N., Le Saux, B., and Lefèvre, S. (2018a). Beyond rgb: Very high resolution urban remote sensing with multimodal deep networks. *ISPRS journal of photogrammetry and remote sensing*, 140:20–32. 43

Audebert, N., Le Saux, B., and Lefèvre, S. (2018b). Generative adversarial networks for realistic synthesis of hyperspectral samples. In *IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium*, pages 4359–4362. IEEE. 36, 48

Audebert, N., Le Saux, B., and Lefèvre, S. (2019). Deep learning for classification of hyperspectral data: A comparative review. *IEEE geoscience and remote sensing magazine*, 7(2):159–173. xv, 41, 45, 46

Badrinarayanan, V., Kendall, A., and Cipolla, R. (2017). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495. 43

Bausch, W. and Khosla, R. (2010). Quickbird satellite versus ground-based multi-spectral data for estimating nitrogen status of irrigated maize. *Precision Agriculture*, 11:274–290. 44

Belkin, M., Niyogi, P., and Sindhwani, V. (2006). Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of machine learning research*, 7(11). 49

Breiman, L. (2001). Random forests. *Machine learning*, 45:5–32. 36

Castillo-Navarro, J., Le Saux, B., Boulch, A., Audebert, N., and Lefèvre, S. (2021). Semi-supervised semantic segmentation in earth observation: The minifrance suite, dataset analysis and multi-task network study. *Machine Learning*, pages 1–36. 41, 50, 51

Cauchy, A. et al. (1847). Méthode générale pour la résolution des systemes d'équations simultanées. *Comp. Rend. Sci. Paris*, 25(1847):536–538. 47

Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., and Abbeel, P. (2016). Infogan: Interpretable representation learning by information maximizing generative adversarial nets. *Advances in neural information processing systems*, 29. 49

Cheng, T., Riaño, D., Koltunov, A., Whiting, M. L., Ustin, S. L., and Rodriguez, J. (2013). Detection of diurnal variation in orchard canopy water content using modis/aster airborne simulator (master) data. *Remote Sensing of Environment*, 132:1–12. 44

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20:273–297. 35

Cover, T. and Hart, P. (1967). Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1):21–27. 36

Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893 vol. 1. 41

Damelin, S. B. and Miller, Jr, W. (2011). *The Mathematics of Signal Processing*. Cambridge University Press. 41

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*. 41, 42

Derksen, D., Inglada, J., and Michel, J. (2020). Geometry aware evaluation of handcrafted superpixel-based features and convolutional neural networks for land cover mapping using satellite imagery. *Remote Sensing*, 12(3). 45

Doersch, C. (2016). Tutorial on variational autoencoders. *arXiv preprint arXiv:1606.05908*. 37, 38

Erhan, D., Courville, A., Bengio, Y., and Vincent, P. (2010). Why does unsupervised pre-training help deep learning? In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 201–208. JMLR Workshop and Conference Proceedings. 49

Estoque, R. C. and Murayama, Y. (2015). Classification and change detection of built-up lands from landsat-7 etm+ and landsat-8 oli/tirs imageries: A comparative assessment of various spectral indices. *Ecological indicators*, 56:205–217. 44

Fix, E. and Hodges, J. L. (1989). Discriminatory analysis. nonparametric discrimination: Consistency properties. *International Statistical Review/Revue Internationale de Statistique*, 57(3):238–247. 36

Gabor, D. (1946). Theory of communication. j inst electr eng–part iii. *Radio Commun Eng*, 93:429–457. 41

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. http://www.deeplearningbook.org. 30, 33, 42

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems*, 27. 41, 49

Granlund, G. H. (1978). In search of a general picture processing operator. *Computer Graphics and Image Processing*, 8(2):155–173. 41

Hao, S., Zhou, Y., and Guo, Y. (2020). A brief survey on semantic segmentation with deep learning. *Neurocomputing*, 406:302–321. 41

Hilt, D. E. and Seegrist, D. W. (1977). *Ridge, a computer program for calculating ridge regression estimates*. Department of Agriculture, Forest Service, Northeastern Forest Experiment 48

Hornik, K., Stinchcombe, M., and White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366. 36

Hu, W., Huang, Y., Wei, L., Zhang, F., and Li, H. (2015). Deep convolutional neural networks for hyperspectral image classification. *Journal of Sensors*, 2015:1–12. 46

Huete, A. R. (1988). A soil-adjusted vegetation index (savi). *Remote sensing of environment*, 25(3):295–309. 44

Izmailov, P., Kirichenko, P., Finzi, M., and Wilson, A. G. (2020). Semi-supervised learning with normalizing flows. In *International Conference on Machine Learning*, pages 4615–4630. PMLR. 49

Jebara, T. (2012). *Machine learning: discriminative and generative*, volume 755. Springer Science & Business Media. 30

Kiefer, J. and Wolfowitz, J. (1952). Stochastic estimation of the maximum of a regression function. *The Annals of Mathematical Statistics*, pages 462–466. 47

Kingma, D. P., Mohamed, S., Jimenez Rezende, D., and Welling, M. (2014). Semi-supervised learning with deep generative models. *Advances in neural information processing systems*, 27. 39, 49

Kingma, D. P. and Welling, M. (2014). Auto-encoding variational bayes. In Bengio, Y. and LeCun, Y., editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*. 37, 38

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. In *Proceedings of the Neural Information Processing Systems (NIPS)*, pages 1097–1105. 41, 42

Laine, S. and Aila, T. (2016). Temporal ensembling for semi-supervised learning. In *International Conference on Learning Representations*. 50

Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324. 41

Li, Y., Zhang, H., and Shen, Q. (2017). Spectral–spatial classification of hyperspectral imagery with 3d convolutional neural network. *Remote Sensing*, 9(1):67. 45, 46

Loève, M. (1977). Elementary probability theory. *Probability Theory I*. 30

Long, J., Shelhamer, E., and Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440. 42

Lowe, D. (1999). Object recognition from local scale-invariant features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 2, pages 1150–1157 vol.2. 41

Luo, W., Li, Y., Urtasun, R., and Zemel, R. (2016). Understanding the effective receptive field in deep convolutional neural networks. *Advances in neural information processing systems*, 29. 45

Miyato, T., Maeda, S.-i., Koyama, M., and Ishii, S. (2018). Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1979–1993. 50

Murphy, K. P. (2012). *Machine learning: a probabilistic perspective*. MIT press. 30

Parzen, E. (1962). On Estimation of a Probability Density Function and Mode. *The Annals of Mathematical Statistics*, 33(3):1065 – 1076. 36

Peña-Barragán, J. M., López-Granados, F., Jurado-Expósito, M., and García-Torres, L. (2007). Mapping ridolfia segetum patches in sunflower crop using remote sensing. *Weed Research*, 47(2):164–172. 44

Ranzato, M. and Szummer, M. (2008). Semi-supervised learning of compact document representations with deep networks. In *Proceedings of the 25th international conference on Machine learning*, pages 792–799. 50

Reynolds, D. A. et al. (2009). Gaussian mixture models. *Encyclopedia of biometrics*, 741(659-663). 36

Rezende, D. and Mohamed, S. (2015). Variational inference with normalizing flows. In *International conference on machine learning*, pages 1530–1538. PMLR. 49

Rifai, S., Mesnil, G., Vincent, P., Muller, X., Bengio, Y., Dauphin, Y., and Glorot, X. (2011). Higher order contractive auto-encoder. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 645–660. Springer. 50

Robbins, H. and Monro, S. (1951). A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407. 47

Rokach, L. and Maimon, O. (2005). Top-down induction of decision trees classifiers - a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 35(4):476–487. 36

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer. 42

Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386. 35

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1985). Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science. 35

Ruthotto, L. and Haber, E. (2021). An introduction to deep generative modeling. *GAMM-Mitteilungen*, 44(2):e202100008. 36, 37

Salah, R., Vincent, P., Muller, X., et al. (2011). Contractive auto-encoders: Explicit invariance during feature extraction. In *Proc. of the 28th International Conference on Machine Learning*, pages 833–840. 49

Santosa, F. and Symes, W. W. (1986). Linear inversion of band-limited reflection seismograms. *SIAM Journal on Scientific and Statistical Computing*, 7(4):1307–1330. 44, 48

Sietsma, J. and Dow, R. J. (1991). Creating artificial neural networks that generalize. *Neural networks*, 4(1):67–79. 48

Sobel, I. (2014). An Isotropic 3x3 Image Gradient Operator. *Presentation at Stanford A.I. Project 1968*. 41

Sollich, P. (1999). Probabilistic methods for support vector machines. *Advances in neural information processing systems*, 12. 35

Spurr, A., Aksan, E., and Hilliges, O. (2017). Guiding infogan with semi-supervision. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 119–134. Springer. 49

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958. 48

Stoian, A., Poulain, V., Inglada, J., Poughon, V., and Derksen, D. (2019). Land cover maps production with high resolution satellite image time series and convolutional neural networks: Adaptations and limits for operational systems. *Remote Sensing*, 11(17):1986. 43

Tipping, M. E. and Bishop, C. M. (1999). Mixtures of probabilistic principal component analyzers. *Neural computation*, 11(2):443–482. 44

Touvron, H., Vedaldi, A., Douze, M., and Jégou, H. (2019). Fixing the train-test resolution discrepancy. *Advances in neural information processing systems*, 32. 41

Triguero, I., García, S., and Herrera, F. (2015). Self-labeled techniques for semi-supervised learning: taxonomy, software and empirical study. *Knowledge and Information systems*, 42(2):245–284. 49

Van der Maaten, L. and Hinton, G. (2008). Visualizing data using t-sne. *Journal of machine learning research*, 9(11). 44

Van Engelen, J. E. and Hoos, H. H. (2020). A survey on semi-supervised learning. *Machine Learning*, 109(2):373–440. 49, 50

Vopson, M. M. (2021). Estimation of the information contained in the visible matter of the universe. *AIP Advances*, 11(10):105317. 30

Wang, J., Perez, L., et al. (2017). The effectiveness of data augmentation in image classification using deep learning. *Convolutional Neural Networks Vis. Recognit*, 11(2017):1–8. 48

Weston, J., Ratle, F., Mobahi, H., and Collobert, R. (2012). Deep learning via semi-supervised embedding. In *Neural networks: Tricks of the trade*, pages 639–655. Springer. 50

Weston, J. and Watkins, C. (1998). Multi-class support vector machines. Technical report, Citeseer. 35

Wu, L., Ma, C., et al. (2018). How sgd selects the global minima in over-parameterized learning: A dynamical stability perspective. *Advances in Neural Information Processing Systems*, 31. 47

Xu, H. (2006). Modification of normalised difference water index (ndwi) to enhance open water features in remotely sensed imagery. *International journal of remote sensing*, 27(14):3025–3033. 44

Xu, H. (2008). A new index for delineating built-up land features in satellite imagery. *International journal of remote sensing*, 29(14):4269–4276. 44

Zha, Y., Gao, J., and Ni, S. (2003). Use of normalized difference built-up index in automatically mapping urban areas from tm imagery. *International journal of remote sensing*, 24(3):583–594. 44

Zhao, W., Chen, X., Chen, J., and Qu, Y. (2020). Sample generation with self-attention generative adversarial adaptation network (sagaan) for hyperspectral image classification. *Remote Sensing*, 12(5):843. 36

Chapter 3

Materials & Methods

Contents

In this section, we first present the hyperspectral images that were used in this thesis. Then, we present the common metrics and validation methods for discriminative and generative machine learning models, as well as some technical specificities related to remote sensing data.

1 Materials: hyperspectral images

A few public hyperspectral images, provided with a partial ground truth of the land cover, are commonly used in the community to evaluate semantic segmentation models. Often, they are limited to 20 classes and few hundreds of pixels at most. Before and in the course of the thesis, hyperspectral images were also acquired by ONERA over Toulouse and its surroundings. Those images are much larger than the open-source hyperspectral images but were not provided with ground truths.

1.1 Public hyperspectral images

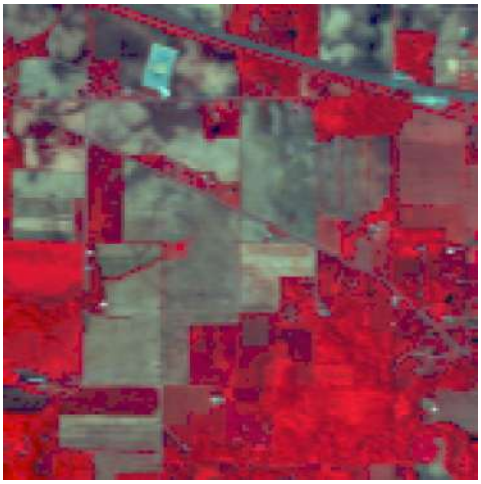
Most hyperspectral images used for semantic mapping in the literature are reflectance images, though some open-source hyperspectral images are provided as radiance images. Radiance images give the averaged reflected radiant flux on each pixel surface while reflectance images give the averaged ratio of the reflected radiant flux on the incident radiant flux. Because reflectance only depends on the matter chemical composition, reflectance images are usually preferred for semantic segmentation. In this thesis, we assume that surfaces are lambertian, *i.e.* that reflectance does not depend on the angle of the incident and the reflected fluxes. We characterize images by their ground sampling distance (GSD), *i.e.* their spatial resolution, their size (number of pixels), their spectral resolution and their spectral range. We present three public hyperspectral data sets, in order of size and spectral diversity, that are commonly used by the community to benchmark segmentation models, and that we use in our own experiments.

Indian Pines is a classic benchmark hyperspectral image acquired by the sensor AVIRIS (Airborne Visible / Infrared Imaging Spectrometer) which covers the spectral domain from 0.4 µm to 2.5 µm with 200 bands at 5 nm spectral resolution and a 20 m GSD. The data is provided as a radiance hypercube whose water absorption bands have been removed (wavelengths for which the radiant flux has been absorbed during its downward and upward path by the water vapor in the atmosphere). We normalize the radiance data such that the maximum value is 1 and the minimum is 0. Indian Pines mainly contains various crops (listed in Tab. 3.1, of which approximately 11,000 pixels are labeled, with very similar spectral signatures and high intra-class variability. A false-color image, the ground truth and the mean spectra per class are shown in Fig. 3.1.

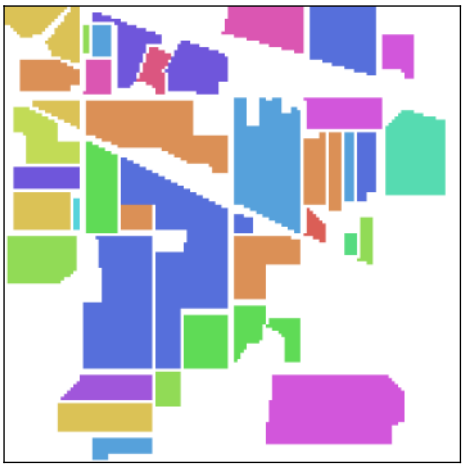
Pavia University is one of the main reference hyperspectral data set acquired by the ROSIS sensor (see Fig. 3.2). It covers the spectral domain from 0.43 µm to 0.86 µm with 103 bands at a 4 nm spectral resolution and a 1.3 m GSD. Atmospheric correction was applied on the image to convert the radiance hypercube to reflectance. It contains 9 classes of vegetation and artificial materials, listed in Tab. 3.2. Some classes such as Bitumen and Asphalt are very similar, as shown in Fig. 3.2. Approximately 42,000 pixels are labeled among 610×340 pixels.

Houston University is, to our knowledge, the largest and most complex public hyperspectral data set. It was acquired by ITRES CASI 1500 sensor that covers the spectral domain from 0.380 µm to 1.05 µm with 48 bands at a 3.5 nm spectral resolution and at a 1 m GSD. More than 400,000 pixels were labeled, split into 20 classes, listed in Tab. 3.3 whose mean spectra are shown in Fig. 3.3: many classes spectrally look very similar. The data report[1] says that a radiometric calibration and correction was applied to convert 'the raw digital numbers into units of spectral radiance'. The available data though are more like reflectance data. It is unclear however which correction was applied as the data lies beyond 1. Consequently, we decided to normalize the data between 0 and 1.

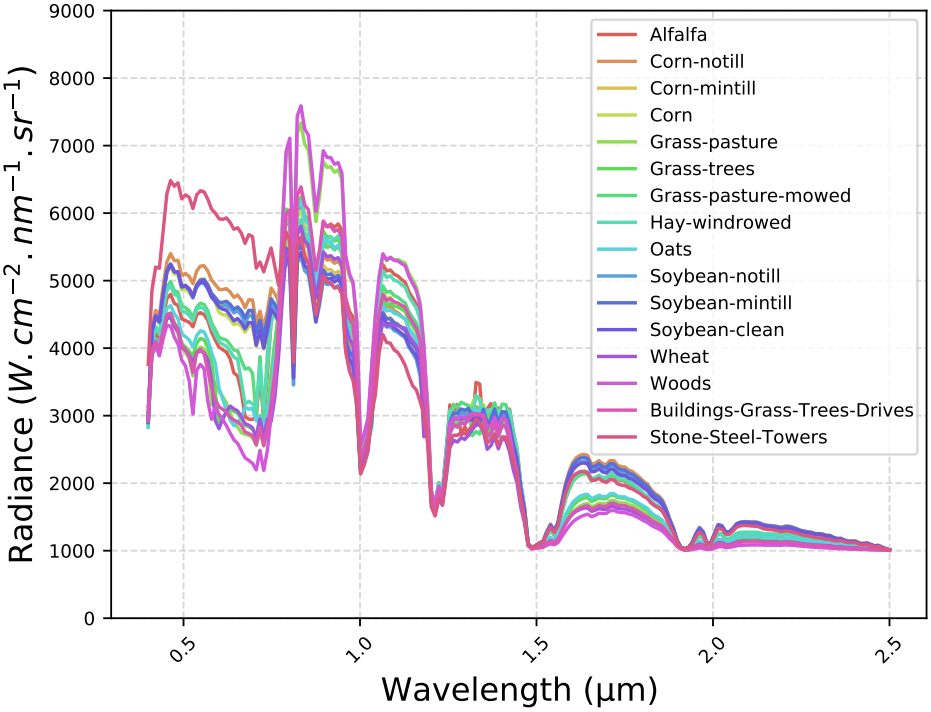
[1] https://hyperspectral.ee.uh.edu/2018IEEEDocs/DataReport.pdf



(a) False-color composition



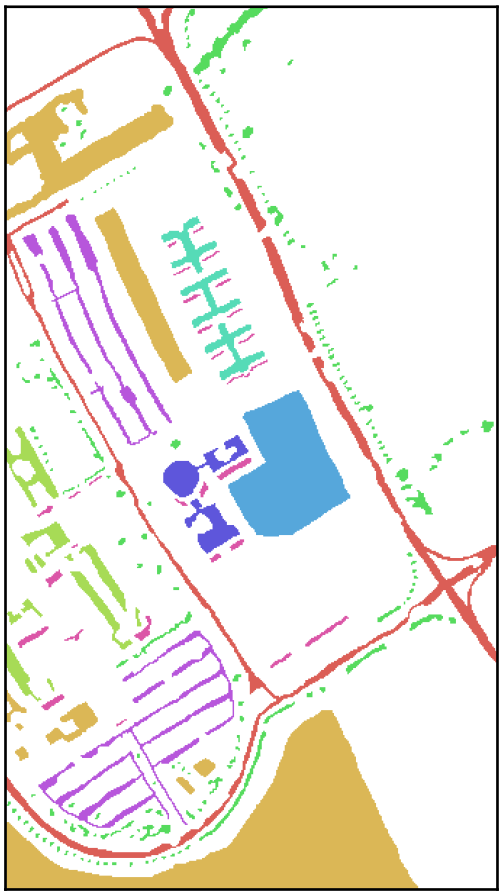
(b) Ground truth



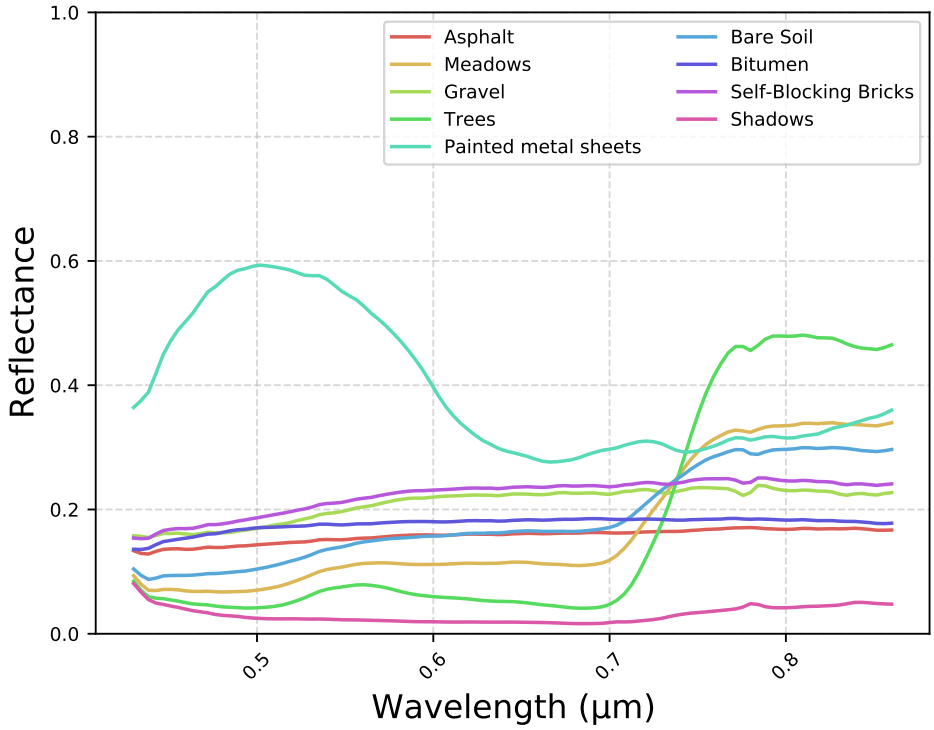
(c) Mean spectra per class

Figure 3.1: Image, ground truth and spectra of the Indian Pines data set

(a) False-color composition



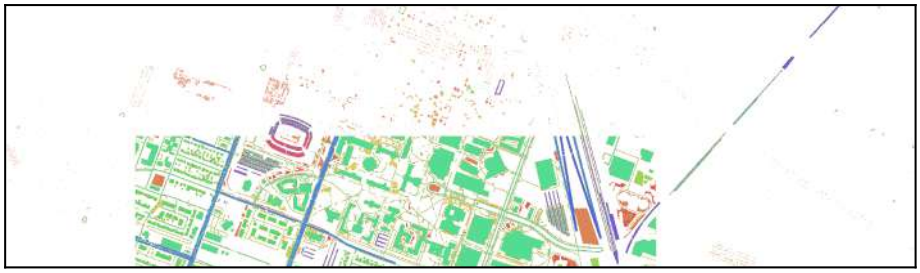
(b) Ground truth



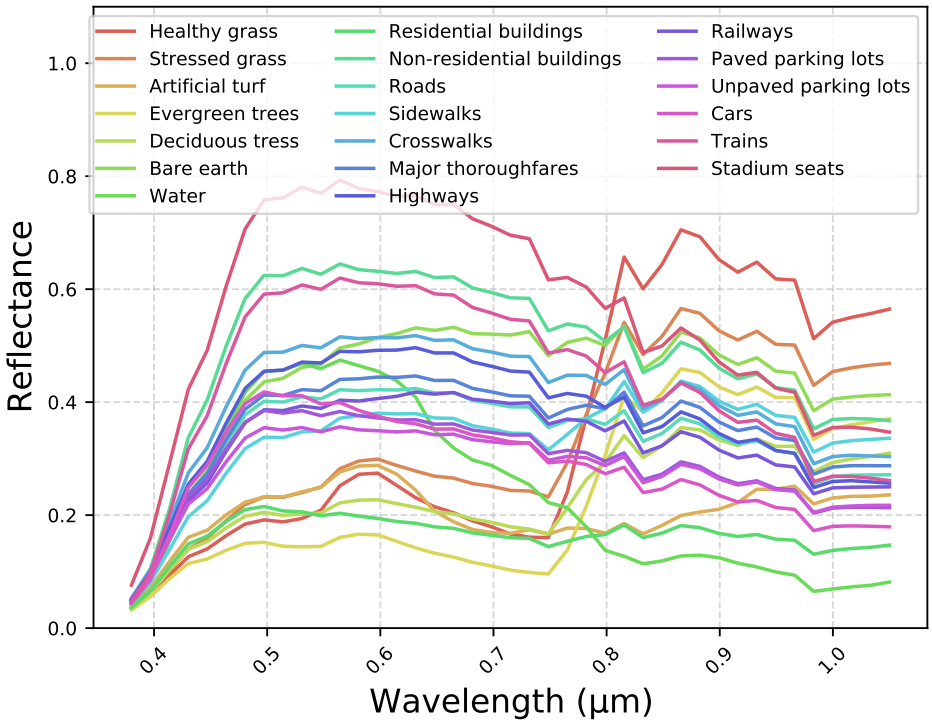
(c) Mean spectra per class

Figure 3.2: Image, ground truth and spectra of the Pavia University data set

(a) False-color composition



(b) Ground truth



(c) Mean spectra per class

Figure 3.3: Image, ground truth and spectra of the Houston University data set

Table 3.1: Classes and number of labeled pixels of the Indian Pines data set

Class id	Class label	Samples
1	Alfalfa	46
2	Corn-notill	1,428
3	Corn-mintill	830
4	Corn	237
5	Grass-pasture	483
6	Grass-trees	730
7	Grass-pasture-mowed	28
8	Hay-windrowed	478
9	Oats	20
10	Soybean-notill	972
11	Soybean-mintill	2,455
12	Soybean-clean	593
13	Wheat	205
14	Woods	1,265
15	Building-Grass-Trees-Drives	386
16	Stone-Steel-Towers	93

Table 3.2: Classes and number of labeled pixels of the Pavia University data set

Class id	Class label	Samples
1	Self-blocking bricks	3,682
2	Meadows	18,649
3	Gravel	2,099
4	Shadows	947
5	Bitumen	1,330
6	Bare soil	5,029
7	Painted metal sheets	1,345
8	Asphalt	6,631
9	Trees	3,064

Table 3.3: Classes and number of labeled pixels of the Houston University data set

Class id	Class label	Samples
1	Healthy grass	9,927
2	Stressed grass	32,585
3	Artificial turf	2,425
4	Evergreen trees	16,419
5	Deciduous trees	9,398
6	Bare earth	6,846
7	Water	673
8	Residential buildings	38,271
9	Non-residential buildings	221,147
10	Roads	41,214
11	Sidewalks	28,841
12	Crosswalks	2,570
13	Major thoroughfares	44,956
14	Highways	9,696
15	Railways	9,745
16	Paved parking lots	11,623
17	Unpaved parking lots	957
18	Cars	4,977
19	Trains	8,596
20	Stadium seats	11,782

1.2 AI4GEO Toulouse hyperspectral image

In the context of the AI4GEO consortium, a hyperspectral image was acquired over the city of Toulouse the 15^{th} of June 2021 around 11am UTC with a AisaFENIX 1K camera (which has a spectral range from 0.4 µm to 2.5 µm with a 3.5 nm spectral resolution in the VNIR and a 12 nm specral resolution in the SWIR, a swath of 1024 m and a GSD of 1 m) that was on-board a Safire aircraft that flew at 1500 m above the ground level. The hyperspectral data was converted in radiance at aircraft level through radiometric and geometric corrections. Then, the radiance image was converted to surface reflectance with the atmospheric correction algorithm COCHISE Miesch et al. [2005]. Hyperspectral surface reflectances were also acquired on-ground with three ASD spectrometers in the range of 0.4 µm to 2.5 µm. For a full description of the data acquired in the CAMCATT / AI4GEO campaign, we refer the reader to the data article [Roupioz et al., 2023]. In preparation of the AI4GEO campaign, a hyperspectral image was acquired with the same camera in 2019 over the cities of Mauzac and Fauga, in the South of Toulouse. In order to test segmentation techniques in life-like scenarios, we partially annotated the Fauga-Mauzac image and the Toulouse image (whose locations are shown in Fig. 3.4), which resulted in three data sets. The data sets, built for different purposes, will be presented and discussed in details in the corresponding chapters.

Figure 3.4: Area of Toulouse over which the hyperspectral image was acquired during the AI4GEO campaign and location of Toulouse in France

2 Methods

2.1 Metrics

In the following, we define the metrics used in this thesis to evaluate discriminative and generative models.

2.1.1 Classification / segmentation metrics

For classification tasks, accuracy metrics are usually generalized from binary classification tasks for which the data set is composed of *positive* and *negative* samples. For binary tasks, *true positives* (TP) are the number of positive samples correctly classified, *true negatives* (TN) are the number of negative samples correctly classified, *false positives* (FP) are the number of negative samples classified as positive and *false negatives* (FN) are the number of positive samples classified as negative. For a multi-class problem and for a class $k \in \{1, \dots, c\}$, positive samples are samples that belong to class k and negative samples are all others. The number of samples / predictions is denoted as N:

$$\forall k \in \{1, \dots, c\}, N = TP_k + TN_k + FP_k + FN_k \tag{3.1}$$

A **confusion matrix** (Q) is a $c \times c$ matrix that indicates the number of confusions between each class: the value Q_{ij} at the i^{th} row and the j^{th} column is equal to the number of samples of class i that are predicted as class j. For instance, Fig. 3.5 shows a confusion matrix for a 3-class problem: 10 samples of class #1 were correctly classified, 3 samples of class #1 were classified as class #2 and 7 samples of class #1 were classified as class #3.

Overall accuracy (OA) is defined as follows:

$$OA = \frac{1}{N} \sum_{k=1}^{c} TP_k \tag{3.2}$$

Precision (P), for a class k, measures which proportions of positive predictions were actually positive:

$$P_k = \frac{TP_k}{TP_k + FP_k} \tag{3.3}$$

Recall (R), for a class k, measures which proportions of positive samples were correctly predicted:

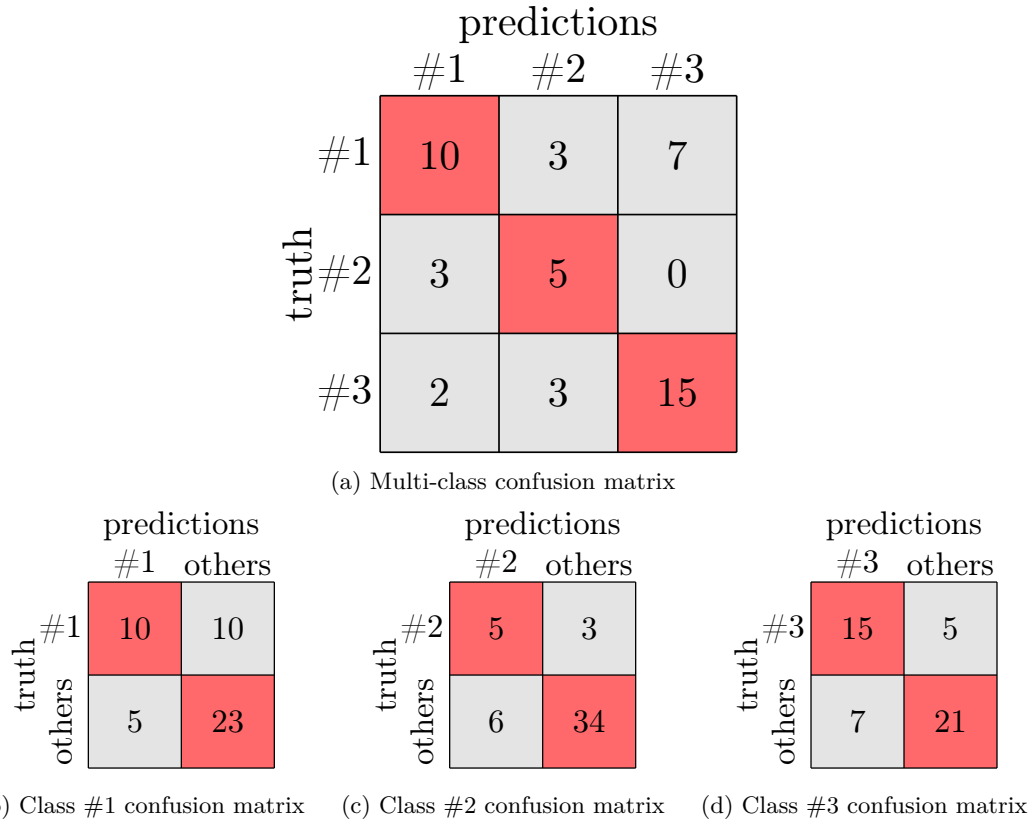
$$R_k = \frac{TP_k}{TP_k + FN_k} \tag{3.4}$$

F1 score (F1), for a class k, is a combination of precision and recall, equal to 1 if both are equal to 1, and equal to 0 if one of them equals 0:

$$F1_k = 2\frac{P_k \cdot R_k}{P_k + R_k} \tag{3.5}$$

Global F1 score is the average of class-wise F1 scores:

$$F1 = \frac{1}{c} \sum_{k=1}^{C} F1_k \tag{3.6}$$



(a) Multi-class confusion matrix

(b) Class #1 confusion matrix (c) Class #2 confusion matrix (d) Class #3 confusion matrix

Figure 3.5: Illustration of a confusion matrix for a 3-class classification problem

Intersection over Union (IoU), for a class k, is the intersection of predictions and labels over the union of predictions and labels:

$$IoU_k = \frac{TP_k}{TP_k + FN_k + FP_k} \tag{3.7}$$

The mean IoU is the average of class-wise IoU:

$$IoU = \frac{1}{c} \sum_{k=1}^{c} IoU_k \tag{3.8}$$

2.1.2 Disentanglement metrics

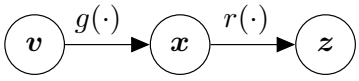


Figure 3.6: Illustration of the generative process $g(\boldsymbol{v}) \longmapsto \boldsymbol{x}$ and of the representation mapping $r(\boldsymbol{x}) \longmapsto \boldsymbol{z}$, where $\boldsymbol{v}$ are the causes, or the factors of variations, $\boldsymbol{x}$ are the observations and $\boldsymbol{z}$ are the latent variables. Figure reproduced from [Carbonneau et al., 2022].

In generative modeling with latent variable probabilistic models, one expected property is disentanglement. A latent space is said to be disentangled if its variables independently capture true underlying factors that explain the data [Carbonneau et al., 2022]. For instance, the factors of variation of hand-written digits $\boldsymbol{x}$ could be the number, the thickness and the orientation of the digit. In the formalism introduced in [Carbonneau et al., 2022], each observation $\boldsymbol{x}^{(i)}$, $i \in \{1, \ldots, N\}$, is assumed to be completely explained by a set of factors $\boldsymbol{v}^{(i)}$, as illustrated in Fig. 3.6. Assessing whether a representation is disentangled is still an

active research topic. Nevertheless, Carbonneau et al. [2022] made a review of state-of-the art metrics to measure disentanglement, that they decompose in modularity, compactness and explicitness. In particular, we are interested here in disentanglement metrics that assume that the true factors $\{\boldsymbol{v}^{(1)}, \dots, \boldsymbol{v}^{(N)}\}$ are known. For the sake of explanation, we consider the following process:

$$r\left(g\left(\begin{bmatrix}\boldsymbol{v}_1\\ \boldsymbol{v}_2\end{bmatrix}\right)\right) = \begin{bmatrix}\boldsymbol{z}_1\\ \boldsymbol{z}_2\\ \boldsymbol{z}_3\end{bmatrix} \tag{3.9}$$

Modularity guarantees that a variation in one factor only affects a subspace of the latent space, and that this subspace is only affected by one factor. An illustration of modularity is the following:

$$r\left(g\left(\begin{bmatrix}\boldsymbol{v}_1 + \Delta\boldsymbol{v}_1\\ \boldsymbol{v}_2\end{bmatrix}\right)\right) = \begin{bmatrix}\boldsymbol{z}_1 + \Delta\boldsymbol{z}_1\\ \boldsymbol{z}_2 + \Delta\boldsymbol{z}_2\\ \boldsymbol{z}_3\end{bmatrix} \tag{3.10}$$

$$r\left(g\left(\begin{bmatrix}\boldsymbol{v}_1\\ \boldsymbol{v}_2 + \Delta\boldsymbol{v}_2\end{bmatrix}\right)\right) = \begin{bmatrix}\boldsymbol{z}_1\\ \boldsymbol{z}_2\\ \boldsymbol{z}_3 + \Delta\boldsymbol{z}_3\end{bmatrix} \tag{3.11}$$

Compactness relates to the size of the subspace affected by the variation in one factor. Finally, **explicitness** expresses how explicit is the relation between the latent code and the factors. The most explicit relation would be a linear relation:

$$r\left(g\left(\begin{bmatrix}\boldsymbol{v}_1\\ \boldsymbol{v}_2 + \Delta\boldsymbol{v}_2\end{bmatrix}\right)\right) = \begin{bmatrix}\boldsymbol{z}_1\\ \boldsymbol{z}_2\\ \boldsymbol{z}_3 + \Delta\boldsymbol{v}_2\end{bmatrix} \tag{3.12}$$

Mutual Information Gap (MIG) [**Chen et al., 2018**]. For one dimension of the factors, $\boldsymbol{v}_i$, the MIG computes the mutual information $I(\boldsymbol{v}_i, \boldsymbol{z}_j)$ between $\boldsymbol{v}_i$ and each dimension of the latent variable $\boldsymbol{z}_j$. The latent code with the highest mutual information is denoted as z_* while the second highest is denoted as $z_\circ$. Then, the MIG is defined as the difference of the highest and the second highest mutual information normalized by the entropy of the factor:

$$MIG(\boldsymbol{v}_i) = \frac{I(\boldsymbol{v}_i, z_*) - I(\boldsymbol{v}_i, \boldsymbol{z}_\circ)}{H(\boldsymbol{v}_i)} \tag{3.13}$$

The global MIG score is the average of individual MIG scores.

Therefore, the MIG is a measure of compactness: a high MIG ensures that one factor is expressed in one dimension of the latent code only. However, one dimension of the latent code could contain information about several factors [Carbonneau et al., 2022].

Joint Entropy Minus Mutual Information Gap (JEMMIG) [**Do and Tran, 2019**]. To measure modularity, the JEMMIG computes the joint entropy of the factor $\boldsymbol{v}_i$ and its best code $\boldsymbol{z}_*$ minus the MIG:

$$JEMMIG(\boldsymbol{v}_i) = H(\boldsymbol{v}_i, \boldsymbol{z}_*) - I(\boldsymbol{v}_i, \boldsymbol{z}_*) + I(\boldsymbol{v}_i, \boldsymbol{z}_\circ) \tag{3.14}$$

In practice, the joint entropy and the mutual information cannot be computed analytically in most cases. A very common numerical estimation of the mutual information, used in the disentanglement-lib[2], consists in dividing the factor and code spaces in B_v and B_z bins, respectively. Then, the probability $p(\boldsymbol{v} = i)$, $i \in \{1, \dots, B_v\}$, is estimated as the proportion

[2] https://github.com/google-research/disentanglement_lib

of factor samples assigned to the i^{th} bin with respect to the total number of samples N. The distribution of latent codes $p(\boldsymbol{z})$ as well as the joint distribution $p(\boldsymbol{v}, \boldsymbol{z})$ are estimated in the same way. The mutual information estimation is given as follows:

$$\hat{I}(\boldsymbol{v}, \boldsymbol{z}) = \sum_{i=1}^{B_v} \sum_{j=1}^{B_z} p(\boldsymbol{v} = i, \boldsymbol{z} = j) \log \left(\frac{p(\boldsymbol{v} = i, \boldsymbol{z} = j)}{P(\boldsymbol{v} = i) P(\boldsymbol{z} = j)} \right) \tag{3.15}$$

The main limitation of this approximation is its dependence to the choice of the number of bins, as illustrated in Fig. 3.7. In the example of Fig. 3.7a with a low number of bins, we have $p(\boldsymbol{v}_1 = 3, \boldsymbol{z}_1 = 2) = 0.1$, $p(\boldsymbol{v}_1 = 4, \boldsymbol{z}_1 = 2) = 0.1$, $p(\boldsymbol{v}_1 = 4, \boldsymbol{z}_1 = 3) = 0.8$, which leads to a low joint entropy, whereas a high number of bins leads to a high joint entropy as illustrated in Fig. 3.7b. At this time, there are no procedure to choose the right balance between coarse and fine discretization [Carbonneau et al., 2022].

(a) $H(\boldsymbol{v}_1, \boldsymbol{z}_1) = 0.64$ (b) $H(\boldsymbol{v}_1, \boldsymbol{z}_1) = 2.3$

Figure 3.7: Illustration of the influence of the discretization on the estimation of the joint entropy.

Finally, Carbonneau et al. [2022] suggest to normalize the JEMMIG metric between 0 and 1, with 1 meaning high disentanglement:

$$JEMMIG(\boldsymbol{v}_i) = 1 - \frac{H(\boldsymbol{v}_i, \boldsymbol{z}_*) - I(\boldsymbol{v}_i, \boldsymbol{z}_*) + I(\boldsymbol{v}_i, \boldsymbol{z}_\circ)}{H(\boldsymbol{v}_i) + \log B_z} \tag{3.16}$$

2.2 Model validation

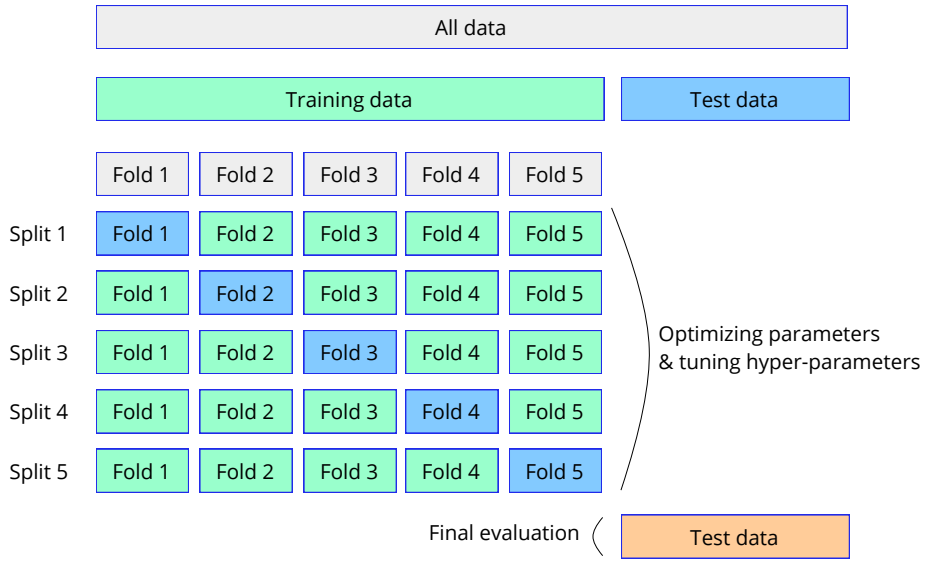


Figure 3.8: Illustration of k-fold cross-validation reproduced and slightly modified from the scikit-learn[1] documentation.

The aim of model validation is to evaluate whether a machine learning model generalizes to unseen data. The generalization performance of the model is computed on a validation set,

whose data shall be independent of the training data. A popular model validation technique is k-fold cross-validation [Stone, 1974]. k-fold consists in randomly dividing the training data in k equal sets. Then, k-1 sets are used for training and the last set is used for validation. In particular, the hyper-parameters of the model, *i.e.* the parameters than cannot be optimized, such as the learning rate of the gradient descent algorithm, the number of epochs or the regularization coefficient, are tuned such that the accuracy of the model on the validation set is maximized. Because the performance of the model could be dependent of the choice of the validation set, k different training and validation splits are done, as illustrated in Fig. 3.8.

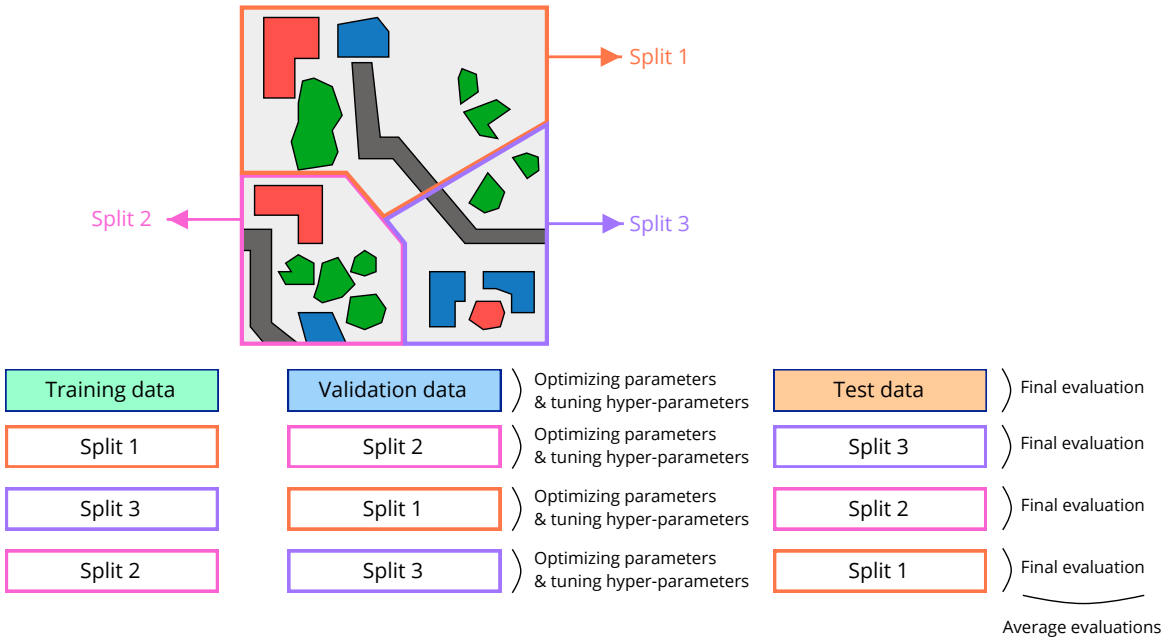


Figure 3.9: Illustration of a common validation process in remote sensing. In general though, the train, validation and test sets have different sizes and different splits are computed.

For semantic segmentation though, random splits of the data $\{(\boldsymbol{x}^{(1)}, y^{(1)}), \ldots, (\boldsymbol{x}^{(N)}, y^{(N)})\}$ in training and validation sets is not appropriate because neighboring pixels are not independent. Therefore, spatially disjoint splits of the data are mandatory for model validation, as argued in several articles [Audebert et al., 2019; Geiß et al., 2017; Karasiak, 2020]. Because a particular spatially disjoint split of the test data set could be in favor of a specific model, it is common to make different train / validation / test splits and average the test metrics, as illustrated in Fig. 3.9.

[1] https://scikit-learn.org/stable/modules/cross_validation.html

3 References

Audebert, N., Le Saux, B., and Lefèvre, S. (2019). Deep learning for classification of hyperspectral data: A comparative review. *IEEE geoscience and remote sensing magazine*, 7(2):159–173. 69

Carbonneau, M.-A., Zaidi, J., Boilard, J., and Gagnon, G. (2022). Measuring disentanglement: A review of metrics. *IEEE Transactions on Neural Networks and Learning Systems*. 66, 67, 68

Chen, R. T. Q., Li, X., Grosse, R. B., and Duvenaud, D. K. (2018). Isolating sources of disentanglement in variational autoencoders. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc. 67

Do, K. and Tran, T. (2019). Theory and evaluation metrics for learning disentangled representations. In *International Conference on Learning Representations*. 67

Geiß, C., Pelizari, P. A., Schrade, H., Brenning, A., and Taubenböck, H. (2017). On the effect of spatially non-disjoint training and test samples on estimated model generalization capabilities in supervised classification with spatial features. *IEEE Geoscience and Remote Sensing Letters*, 14(11):2008–2012. 69

Karasiak, N. (2020). Museo toolbox: A python library for remote sensing including a new way to handle rasters. *Journal of Open Source Software*, 5(48):1978. 69

Miesch, C., Poutier, L., Achard, V., Briottet, X., Lenot, X., and Boucher, Y. (2005). Direct and inverse radiative transfer solutions for visible and near-infrared hyperspectral imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 43(7):1552–1562. 64

Roupioz, L., Briottet, X., Adeline, K., Al Bitar, A., Barbon-Dubosc, D., Barda-Chatain, R., Barillot, P., Bridier, S., Carroll, E., Cassante, C., Cerbelaud, A., Déliot, P., Doublet, P., Dupouy, P., Gadal, S., Guernouti, S., De Guilhem De Lataillade, A., Lemonsu, A., Llorens, R., Luhahe, R., Michel, A., Moussous, A., Musy, M., Nerry, F., Poutier, L., Rodler, A., Riviere, N., Riviere, T., Roujean, J., Roy, A., Schilling, A., Skokovic, D., and Sobrino, J. (2023). Multi-source datasets acquired over toulouse (france) in 2021 for urban microclimate studies during the camcatt/ai4geo field campaign. *Data in Brief*, 48:109109. 64

Stone, M. (1974). Cross-validatory choice and assessment of statistical predictions. *Journal of the royal statistical society: Series B (Methodological)*, 36(2):111–133. 69

Chapter 4

Active Learning: improving hyperspectral image mapping with few additional labeled pixels

Contents

1 Chapter summary

Often, the difficulty of machine learning tasks lies in the variability of data. In the context of classification and segmentation, the intra-class variability is usually so much complex that machine learning models with appropriate architectures are required to learn robust features. Deep convolutional neural networks have become the state of the art both for natural image and RGB remote sensing image segmentation because 1) most of the discriminative information is spatial in these cases and 2) CNNs are very effective in leveraging the information of geometry, texture and context. For hyperspectral data however, most of the discriminative information is spectral. The spectral intra-class variability, that comes along with spectral inter-class similarities, can be casted into three categories: *physics*, *intrinsic* and *semantic* variability. *Semantic* intra-class variability especially can be very large, as far as a great diversity of materials is represented by a few dozen of classes. Therefore, the choice of the nomenclature and the choice of the spectra to represent each class (that is the labeled training data) are of crucial importance. As much as, on the one hand, an airborne hyperspectral image over a metropolis typically contains more than hundreds of millions of pixels with a highly imbalanced distribution of the land cover classes, and that, on the other hand, the field campaigns required to label the data are very expensive, active learning techniques seem imperative to optimally guide the annotation of the ground truth. Active learning (AL) methods are iterative algorithms that select the most informative samples to be labeled, given an initial training data set and a classifier [Settles, 2012]. Then, an oracle, *i.e.* an expert, labels the given pixels that are added to the training set.

Problematic

To what extent can AL techniques, by guiding the annotation of few additional pixels by human experts, can improve the land cover mapping of hyperspectral images?

■ Summary of contributions

⋆ **Synthesis of AL strategies & unification of the AL formalism** In section 2, we suggest to cast active learning techniques in three main categories: uncertainty-based, representativeness-based and performance-based techniques. Uncertainty-based AL methods consider that the most informative pixels are the ones for which the classifier is the most uncertain. Representativeness-based AL methods consider that the most informative pixels are the ones which represent the whole data set at best. Performance-based AL methods select the pixels most likely to reduce the generalization error. In section 3.1, we select (based on publication date and triggered interest in the community) and bring under the same formalism seven state-of-the-art techniques, listed in Tab. 4.1.

Table 4.1: Benchmarked Active Learning techniques

Uncertainty-based	Representativeness-based	Performance-based
[Tong Luo et al., 2004]	[Sener and Savarese, 2018]	[Konyushkova et al., 2017]
[Houlsby et al., 2011]	[Dasgupta and Hsu, 2008]	
[Kirsch et al., 2019]	[Sinha et al., 2019]	

⋆ **Empirical study of AL potential** In order to assess the capacities of the methods to significantly improve the mapping of a hyperspectral image over a metropolis, by labeling only a few pixels, we carry numerical experiments on four data sets in section 3.2, during which

we compute accuracy metrics on the segmentation task and report the number of selected pixels per class over AL iterations. Results highlight the complementarity of AL strategies.

⋆ **Decrease of AL computational requirements by data preprocessing** In order to cope with the computation time of some AL algorithms that hinders their practical use, we introduce different preprocessing techniques and study their impact on the performance of AL in section 4. Preprocessing methods consist in segmenting the image in superpixels, on which the AL technique is applied. Then, pixels to label are sampled from the most informative superpixels.

⋆ **Integration of a priori semantic knowledge** Finally, a drawback of AL that has not been considered in the literature is that AL techniques give equal importance to every classes. However, in remote sensing, and especially for impermeable surfaces segmentation, some confusions are much more problematic than others. Indeed, the greater the permeability difference between two classes, the more serious the confusion. Therefore, we integrate this a priori semantic knowledge in the framework of Breaking Ties to improve the specific task of impermeable surfaces mapping in section 5.

2 Active learning principles

In order to facilitate and guide data annotation, Active Learning (AL) methods have been extensively studied in the machine learning literature. AL techniques aim to iteratively and interactively enrich the training data set with informative samples. The category of AL methods of interest in this thesis is referred as pool-based AL: for a given number of iterations, a query system selects the most informative samples from a pool of unlabeled data and an oracle labels the given samples, which are added to the training set. The query system is also usually called the acquisition function, which gives to each candidate data point a score that reflects how informative it is for the task at hand. The oracle is usually an expert that can annotate the data with confidence. The flowchart of pool-based AL is illustrated in Fig. 4.1. Other AL methods are based on the synthesis of training samples with generative models. [Zhu and Bento, 2017] for instance use an uncertainty criterion to sample informative samples from a generative adversarial network on hand-written digits and natural images. The idea is that the generated samples should bring more diversity to the existing training set. Nevertheless, in our case, it is not possible to label a spectrum without the context given by surrounding pixels. A large neighborhood (at least 10×10 pixels) is needed to distinguish many classes, such as high vegetation from short vegetation or even tile from bare soil. If works have been carried out in the remote sensing community to synthesize hyperspectral spectra with generative models [Audebert et al., 2018; Zhao et al., 2020], synthesizing hyperspectral cubes, to the best of our knowledge, has not yet been achieved. For this reason, we kept generative-based methods out of scope of our work. Moreover, we only considered methods for which the oracle provides the true labels, in contrast with pseudo-labeling frameworks in which some labels are given by a classifier [Zhang et al., 2020]. Pseudo-labeling methods rely on the following hypothesis, that does not hold true in an operational context: confident predictions (when the classifier shows a high probability) are always correct, and therefore can be used as annotations. It has been shown that machine learning models can make wrong predictions with high confidence, especially if the training data set is not representative of the whole image. For this reason, we argue that pseudo-labeling methods are not reliable enough to be used in an operational scenario. Finally, we focused on the task of hyperspectral image segmentation but we believe that our results could be extended to other data such as satellite multispectral imaging or other tasks than classification. For instance, [Ruzicka et al., 2020] used AL methods to enhance change detection algorithms and [Kellenberger et al., 2019]

applied AL algorithms for object detection. In those cases where the information lies in rare and specific samples, we believe that active learning is even more crucial. Typically, using random sampling in the context of change or object detection would yield very poor results as the object of interest would be generally rare in the image.
Active Learning for semantic segmentation can be divided into three approaches: image-level, region-level and pixel-level AL [Xie et al., 2022]. Image-level methods may rely on dense ground truth annotations such as [Xie et al., 2020] that selects difficult to segment images. Though image-level techniques could be used on a hyperspectral image by dividing the image in smaller patches, the use of region-based or pixel-based techniques would be more straightforward. In this study, we mainly focus on pixel-based approaches, which comes down to AL for classification tasks, as far as the spectral dimension is much more informative on the land cover than the spatial dimension.
Acquisition functions are at the core of pool-based AL techniques: they define what is an informative data sample. Previous reviews in 2010 and 2011 suggested different taxonomies [Settles, 2012; Tuia et al., 2011] but based on recent advances that have built on existing heuristics or paved the way toward new approaches, we suggest to class acquisition functions in three main categories: uncertainty, representativeness and performance heuristics.

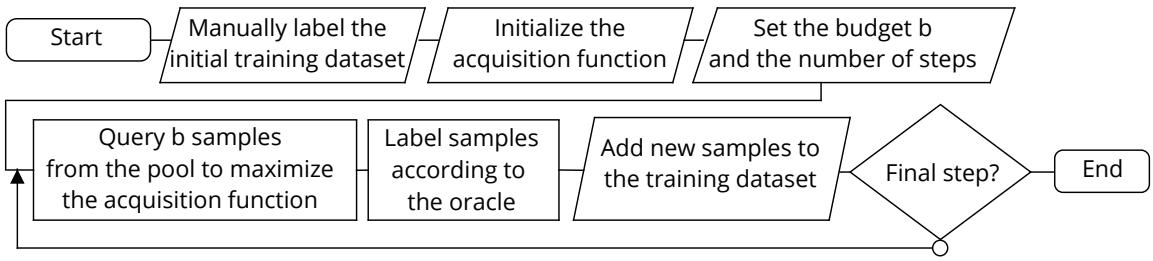


Figure 4.1: Pool-based Active Learning Flowchart

2.1 AL Framework

We denote a data point $\boldsymbol{x} \in \mathcal{X}$ and its class $y \in \{1, ..., C\}$, where $\mathcal{X}$ is the feature space and C is the number of classes. In our case, $\mathcal{X} = [0, 1]^B$ where B is the number of spectral bands. From a data set $\mathcal{D} = \{\boldsymbol{x}_i\}_{i\in\{1,...,N\}}$ of N samples, we manually label an initial training data set $\mathcal{D}_0^{train} = (L_0, Y_0) = \{\boldsymbol{x}_i, y_i\}_{i\in\{1,...,N_0\}}$. The N_t labeled points at step t are denoted as L_t and the associated labels as Y_t. At each step t, we query b unlabeled points $\{\boldsymbol{x}_1^*, \ldots, \boldsymbol{x}_b^*\}$ from the current unlabeled pool $U_t = \mathcal{D}\backslash L_t$ that maximize an acquisition function $a : \mathcal{X}^b \longrightarrow \mathbb{R}$ that is parametrized by $\mathcal{D}_t^{train}$ and U_t. This parametrization will be sometimes left implicit in the following. An oracle provides the true labels $\{y_1, ..., y_b\}$. The active learning process is detailed by Algorithm 1. In the following, we shall denote $\boldsymbol{x}_{1:b} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_b\}$ and $y_{1:b} = \{y_1, ..., y_b\}$.

2.2 Uncertainty-based AL

Uncertainty heuristics rely on machine learning models that make predictions from the data given the training data set. They select data points for which the model is the most uncertain about. They split into conventional inter-class uncertainty heuristics and more recent epistemic uncertainty heuristics. Inter-class uncertainty heuristics select points for which the model hesitates between two classes. They include Breaking Ties [Tong Luo et al., 2004], prediction entropy [Shannon, 1948] or variation ratios [Freeman, 1965], which are highly related to margin-based heuristics such as margin sampling [Ferecatu and Boujemaa,

Algorithm 1: Active Learning Framework

Data: Initial labeled data set L_0; Initial pool U_0;
Inputs: Number of steps n; Budget b;
Acquisition function a;
for $t \leftarrow 1$ ***to*** n **do**
$\{\boldsymbol{x}_1^*, ..., \boldsymbol{x}_b^*\} \leftarrow \underset{\{\boldsymbol{x}_1,...,\boldsymbol{x}_b\} \subset U_t}{\text{argmax}} a(\{\boldsymbol{x}_1, ..., \boldsymbol{x}_b\}; \mathcal{D}_t^{train}, U_t)$;
$\{y_1, ..., y_b\} \leftarrow Oracle(\{\boldsymbol{x}_1^*, ..., \boldsymbol{x}_b^*\})$;
$L_{t+1} \leftarrow L_t \cup \{\boldsymbol{x}_1^*, ..., \boldsymbol{x}_b^*\}$;
$Y_{t+1} \leftarrow Y_t \cup \{y_1, \ldots, y_b\}$;
$U_{t+1} \leftarrow U_t \backslash \{\boldsymbol{x}_1^*, ..., \boldsymbol{x}_b^*\}$;
$\mathcal{D}_{t+1}^{train} \leftarrow (L_{t+1}, Y_{t+1})$;
end
Result: $\mathcal{D}_n^{train}$

2007; Tuia et al., 2011] or multiclass level uncertainty [Demir et al., 2011; Tuia et al., 2011]. Margin-based heuristics compute distances between data points and support vector machine hyperplans. Those heuristics are notably relevant to refine the decision frontiers between similar classes with high intra-class variability. They are not suited, however, to measure epistemic uncertainty, *i.e.* uncertainty in little-known data space regions. An example of early epistemic uncertainty heuristic is the so-called "query by committee" method [Settles, 2012; Seung et al., 1992] which measures the disagreement between a committee of models. It is based on the fact that predictions of independant classifiers diverge in regions without training data. In the remote sensing community, the multiview disagreement method [Di and Crawford, 2010] uses a committee of classifiers trained on uncorrelated views from the feature space (subparts of the feature space). The intuition of multiview is the same as dropout [Hinton et al., 2012] for neural networks: it prevents the co-adaptation of feature detectors. Finally, the recent advances in variational inference [Gal and Ghahramani, 2016; Kingma and Welling, 2014; Ranganath et al., 2014; Wingate and Weber, 2013] have paved the way to modern bayesian models and new epistemic heuristics [Houlsby et al., 2011; Kirsch et al., 2019; Li and Guo, 2013; Yang et al., 2015] that fully exploit the capacity of bayesian models to *know when they do not know* as mentioned in [Gal, 2016]. A convenient way to approximate bayesian models is called Monte Carlo dropout, which was introduced by [Gal and Ghahramani, 2016], that consists in using dropout at test time. In that sense, multiview used with a one-layer neural network could be thought as a bayesian approximation as well, as it is a special case of dropout, where the weights related to one view are zeroed out. All in all, epistemic uncertainty-based AL methods are usually criticized for their high sensibility to outliers that may lead to non-representative data sets.

2.3 Representativeness-based AL

In order to circumvent the problems raised by uncertainty heuristics, representativeness based approaches try to query points in order to fit the unlabeled data distribution to the labeled one. Some methods minimize the maximum mean discrepancy (a measure of distance) between the distributions [Wang and Ye, 2015], while other methods have recently considered active learning as a core-set, or sub-modularity problem [Sener and Savarese, 2018; Wei et al., 2015]. Core-set problems aim at selecting a subset of the training data set, while inducing minimal performance loss. The idea behind those approaches is to query points in order to have labeled points homogeneously spread over the whole data space. Meanwhile, methods such as [Sinha et al., 2019] and [Ducoffe and Precioso, 2018] try to fool a discriminator

that spots unlabeled points from labeled points in an adversarial fashion. Those methods implicitly bridge the gap between the labeled and unlabeled data distributions. Adversarial approaches scale well to large data sets as they benefit from the mini-batch optimization of the discriminator. Finally, a cluster based approach introduced by [Dasgupta and Hsu, 2008] queries points so that every cluster in the data is represented by enough training samples. In remote sensing, a land cover nomenclature can be described in a tree shape with nested subclasses. For instance, a subclass *dry vegetation* could lie inside a bigger class *vegetation*. The more branches the tree has, the more finely the nomenclature distinguishes the classes. The idea of the hierarchical sampling method is to query pixels so that the training data set can accurately capture the level of detail in the nomenclature.

2.4 Performance-based AL

Performance based methods query points that increase the classifier performances on the test set. Those methods include expected error reduction and variance reduction [Settles, 2012] through approximations or closed-form formulations. Those methods estimate how much labeling new samples reduce the prediction error on the test data set. Recently, [Konyushkova et al., 2017] suggested to train a regression model in order to predict the expected error reduction based on the state of the model and the candidate samples. To circumvent the difficulty of estimating the test error reduction, [Baram et al., 2003] uses a reinforcement learning paradigm where each sample is a resource to be chosen. After a resource, *i.e.* a sample, is chosen, it is labeled and added to the training data set. A reward is then given according to the increase of accuracy the new samples have allowed. This is a particular setting of the multi-armed bandit problem with a varying number of resources. Meanwhile, meta-learning methods such as [Hsu and Lin, 2015] tackle the AL problem as a multi-armed bandit problem where the algorithm adaptively learns a querying strategy based on a committee of querying systems. It is referred as a meta-learning method because the active learning algorithm itself is learned. The statement of this method is that the efficiency of AL methods are data-dependent: one query system can be efficient on a peculiar data set while being irrelevant on an other one. However, the goal of the present chapter is to understand how AL methods work individually and in which configurations to use them.

3 Comparative study of state-of-the-art methods

Active Learning has demonstrated large benefits on traditional machine learning tasks, such as natural image classification, but their use on hyperspectral images still raises open questions despite the high interest in the community. We first justify the choice of the seven benchmarked AL techniques and thoroughly summarize their mathematical formulations in a unified formalism in section 3.1. We particularly pay attention to the parameters that rule their behaviors. Then, we present the objectives, the design and the results of the numerical experiments in section 3.2. Finally, we discuss the results and their implications in section 3.3.

3.1 Benchmarked methods

We narrowed our study to seven methods of interest.

· Uncertainty based methods

- **Breaking Ties** [Tong Luo et al., 2004] as it is a state-of-the-art inter-class uncertainty heuristic and has extensively been used in active learning frameworks both in the machine learning and remote sensing literatures,
- **Batch-BALD** [Kirsch et al., 2019] as it is expected to improve the state-of-the-art epistemic uncertainty heuristic **BALD** [Houlsby et al., 2011] by enforcing diversity within a batch. We will also conduct experiments with BALD, which is a particular case of Batch-BALD with much fewer computational requirements,

· Representativeness based methods

- **VAAL** [Sinha et al., 2019] as it introduces a new paradigm based on an adversarial framework to tackle the problem of representativeness,
- **Core-set** [Sener and Savarese, 2018] as it has demonstrated promising results and theoretical guarantees by addressing the active learning problem as a core-set problem,
- **Hierarchical sampling** [Dasgupta and Hsu, 2008] as it is a state-of-the-art cluster-based method that could leverage the natural hierarchical structure of remote sensing data,

· Performance based methods

- **LAL** [Konyushkova et al., 2017] as it has recently introduced a new strategy to approximate the test error reduction that guides some performance based query systems.

Breaking Ties A classifier $f : \mathcal{X} \longrightarrow [0,1]^C$ outputs a probability distribution over C classes from a data point $\boldsymbol{x} \in \mathcal{X}$. The Breaking Ties heuristic for a batch of data points $\boldsymbol{x}_{1:b}$ is a measure of confidence:

$$a_{breaking-ties}(\boldsymbol{x}_{1:b}) = \sum_{k=1}^{b} a_{breaking-ties}(\boldsymbol{x}_k) \tag{4.1}$$

$$= -\sum_{k=1}^{b} \left(\max_{i \in (1,\dots,C)} f(\boldsymbol{x}_k)^i - \max_{j, j \neq i} f(\boldsymbol{x}_k)^j \right) \tag{4.2}$$

where $f(x_k)^i$ denotes the i^{th} coordinate of $f(\boldsymbol{x}_k)$, *i.e.* the estimated probability that $\boldsymbol{x}_k$ belongs to class i. The function $a_{breaking-ties}$ takes high values when a data point lies near a decision boundary. It is thus a reliable metric to spot data points that are highly informative of discriminative features between similar classes.

The strong assumption we make using Breaking Ties is that we can be confident about the labels of data points predicted with probabilities close to 1. Previous works showed that this assumption might not always hold true as machine learning models are usually very bad at extrapolating in little-known regions of the data space [Gal, 2016]. For instance, we can consider, in a remote sensing scenario, a rare but relevant class such as pool covers that would not be present in the initial training data set. The Breaking Ties heuristic may never points towards those points as they would not span nearby a decision frontier. When using Breaking Ties, we simultaneously select the b points that maximize $a_{breaking-ties}$ which might induce redundancy within a batch. Nevertheless, Breaking Ties offers major advantages such as its scalability to large data sets and its small number of hyper-parameters.

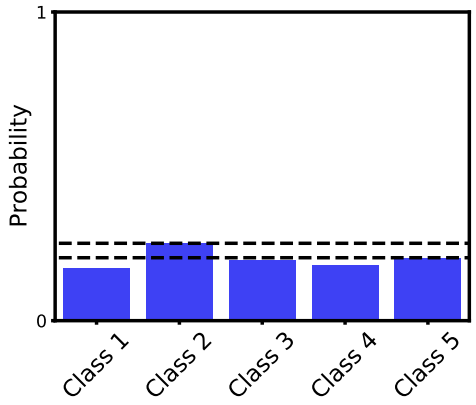


Figure 4.2: Illustration of the Breaking Ties heuristic. The gap between the dotted lines represent the gap between the two most high probabilities.

Batch-BALD A Bayesian model makes predictions $p_{\boldsymbol{\theta}}(y|\boldsymbol{x})$ where the model parameters $\boldsymbol{\theta}$ are conditioned on the training data set $\mathcal{D}_t^{train}$ at step t: $\boldsymbol{\theta} \sim p(\boldsymbol{\theta}|\mathcal{D}_t^{train})$. In other words, the prediction of the label y of a test data point $\boldsymbol{x}$ by a Bayesian model is the expectation of the prediction conditioned on the model parameters:

$$p_{\boldsymbol{\theta}}(y|\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{\theta} \sim p(\boldsymbol{\theta}|\mathcal{D}_t^{train})} p_{\boldsymbol{\theta}}(y|\boldsymbol{x}, \boldsymbol{\theta}) \tag{4.3}$$

Batch-BALD is an epistemic uncertainty function that estimates the mutual information between the model predictions of multiple data points $\{y_1, ..., y_b\}$ and the model parameters:

$$a_{batch-bald}(x_{1:b}; \mathcal{D}_{train}) = I(y_{1:b}; \boldsymbol{\theta} \mid \boldsymbol{x}_{1:b}, \mathcal{D}_{train}) \tag{4.4}$$

We can think of batch-bald as answering the question: how would labelling the subset of points $\{x_1, ..., x_b\}$ inform us on the right probability distribution over the model parameters $\boldsymbol{\theta}$? Another way to consider batch-Bald is to re-write it as:

$$a_{batch-bald}(x_{1:b}; \mathcal{D}_{train}) = H(y_{1:b}|\boldsymbol{x}_{1:b}, \mathcal{D}_{train}) - \mathbb{E}_{p(\boldsymbol{\theta}|\mathcal{D}_{train})}\left[H(y_{1:b}|\boldsymbol{x}_{1:b}, \boldsymbol{\theta}, \mathcal{D}_{train})\right] \tag{4.5}$$

where H denotes the entropy defined in Chapter 2. The first term of the right-hand side of equation 4.5 is high when the model is overall uncertain about its predictions and the second term is low when the model is certain for each draw of model parameters from the posterior $p(\boldsymbol{\theta}|\mathcal{D}_{train})$. **Thus $a_{batch-bald}$ is high when different draws of the model parameters lead to confident but disagreeing predictions**. Batch-BALD is actually an extension of the technique BALD introduced by [Houlsby et al., 2011]. BALD is the special case where the size of the batch is equal to one, which computation is illustrated in Fig. 4.3.

To compute Batch-BALD, we have to compute expectations over the model parameters which we can only approximate using Monte-Carlo samples $\boldsymbol{\theta}_k \sim p(\boldsymbol{\theta}|\mathcal{D}_{train})$. Since every y_i conditioned on $\boldsymbol{\theta}$ are mutually independent, [Kirsch et al., 2019] approximate the second term of the right-hand side of equation 4.5 as follows:

$$\begin{aligned} \mathbb{E}_{p(\boldsymbol{\theta}|\mathcal{D}_{train})} H(y_{1:b}|\boldsymbol{x}_{1:b}, \boldsymbol{\theta}, \mathcal{D}_{train}) &= \mathbb{E}_{p(\boldsymbol{\theta}|\mathcal{D}_{train})} \sum_{i=1}^{b} H(y_i|\boldsymbol{x}_i, \boldsymbol{\theta}, \mathcal{D}_{train}) \\ &\approx \frac{1}{K} \sum_{k=1}^{K} \sum_{i=1}^{b} H(y_i|\boldsymbol{x}_i, \boldsymbol{\theta}_k, \mathcal{D}_{train}) \quad ; \quad \boldsymbol{\theta}_k \sim p(\boldsymbol{\theta}|\mathcal{D}_{train}) \end{aligned} \tag{4.6}$$

The second term is more difficult to compute as the joint probability of $y_{1:b}$ does not factorize

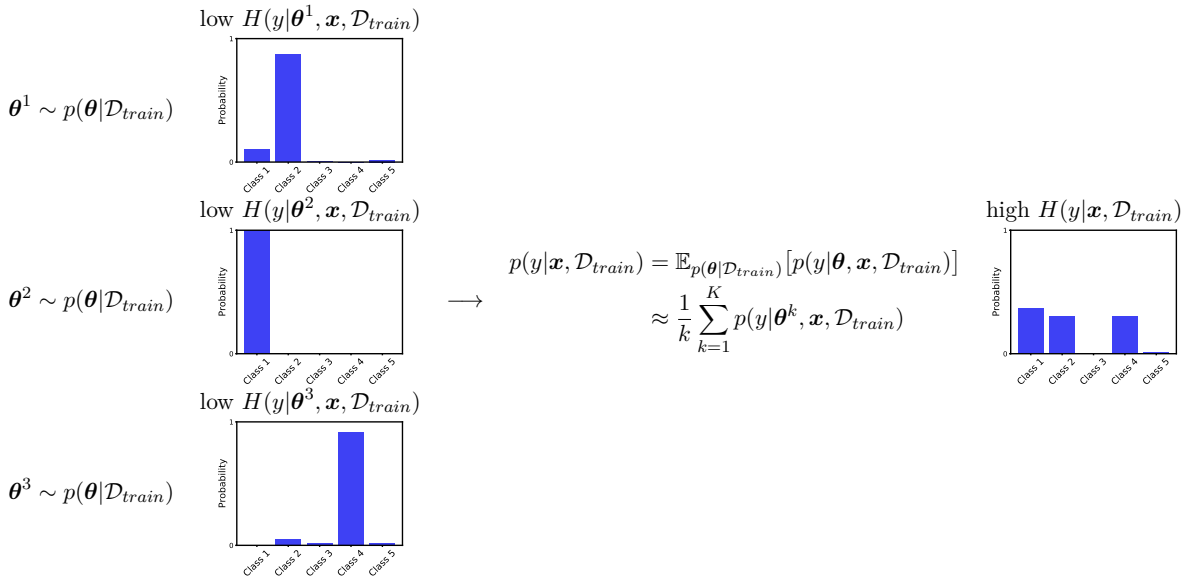


Figure 4.3: Illustration of the computation of the BALD heuristic.

and we have to sum over all possible C^b values of $y_{1:b}$, which we denote as $\mathcal{Y}_{1:b}$:

$$\begin{aligned} H(y_{1:b}|\boldsymbol{x}_{1:b}, \mathcal{D}_{train}) &= \mathbb{E}_{p(y_{1:b}|\boldsymbol{x}_{1:b}, \mathcal{D}_{train})}\big[-\log\, p(y_{1:b}|\boldsymbol{x}_{1:b}, \mathcal{D}_{train})\big] \\ &= \sum_{y_{1:b}\in\mathcal{Y}_{1:b}} p(y_{1:b}|\boldsymbol{x}_{1:b}, \mathcal{D}_{train})\big[-\log\, p(y_{1:b}|\boldsymbol{x}_{1:b}, \mathcal{D}_{train})\big] \\ &\approx -\frac{1}{M}\sum_{m=1}^{M} \log\, p(y^m_{1:b}|\boldsymbol{x}_{1:b}, \mathcal{D}_{train}) \quad ; \quad y^m_{1:b} \sim p(y_{1:b}) \end{aligned} \tag{4.7}$$

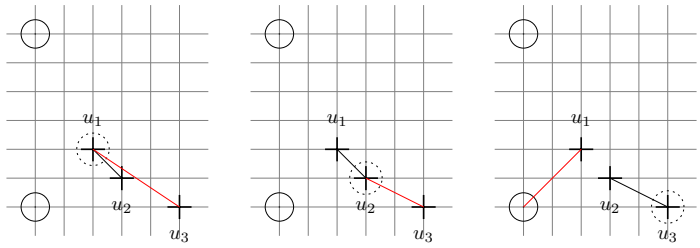
At this point, we can compute $a_{batch-bald}$ for any batch of points. There are however $\binom{|U_t|}{b} = \frac{|U_t|!}{b!(|U_t|-b)!}$ possible batches of points $\{\boldsymbol{x}_1, ..., \boldsymbol{x}_b\} \subset U_t$, which turns the optimization problem intractable. To circumvent this problem, [Kirsch et al., 2019] introduce a greedy algorithm which consists in iteratively querying a data point $\boldsymbol{x}_j$ in order to maximize $a_{batch-bald}(x_{1:t-1} \cup \{\boldsymbol{x}_t\}; \mathcal{D}^{t-1}_{train})$ at step t.

The main disadvantage of batch-bald is that it requires heavy computational resources on large data sets. If the implementation introduced in [Kirsch et al., 2019] makes use of tensor multiplication on GPU, that speeds up the computation of the joint entropy, it is at the expense of heavy memory requirements as a $|U_t| \times M \times C$ matrix is stored, where we recall that $|U_t|$ is the number of unlabeled points at step t, M is the number of samples drawn to estimate the joint entropy $H(y_{1:b}|\boldsymbol{x}_{1:b}, \mathcal{D}_{train})$ and C is the number of classes. Moreover, the larger the batches or the number of classes, the tougher the estimation of the mutual information, as much as the joint entropy is a sum of C^b terms. Compared to Batch-BALD, BALD is much easier to estimate but is more prone to query redundant samples because it only considers samples individually, like Breaking Ties.

Core-set The Core-set technique introduced by [Sener and Savarese, 2018] aims to solve a k-center problem, *i.e.* to select a subset $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_b\}$ of the unlabeled pool U_t such that the largest distance between an unlabeled data point and its nearest labeled point is minimized:

$$a_{core-set}(\boldsymbol{x}_{1:b}; L_t, U_t) = -\max_{\boldsymbol{x}_i \in U_t} \min_{\boldsymbol{x}_j \in x_{1:b} \cup L_t} \Delta(\boldsymbol{x}_i, \boldsymbol{x}_j) \tag{4.8}$$

where Δ is a distance in the classifier output space. The problem is illustrated in Fig. 4.4 on a toy example.



(a) Candidate u_1 (b) Candidate u_2 (c) Candidate u_3

$\circ \in L_t$ $+ \in U_t$ $\oplus \in \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_b\}$

the largest distance between unlabeled points without the candidate points $\boldsymbol{x}_i \in U_t \setminus \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_b\}$ and the labeled points with the candidate points $\boldsymbol{x}_j \in \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_b\} \cup X_l$

Figure 4.4: Illustration of the Core-set technique with a batch size of one. Circles denote labeled points while crosses denote unlabeled points. Lines between unlabeled points and labeled + candidate points are drawn. The largest distance is shown in red while the other is shown in black. The candidate point, among u_1, u_2, u_3, that minimizes the largest distance is u_2. Therefore, the Core-set technique would select u_2 to be labeled in this case.

The idea is that every unlabeled point should be included in the smallest possible balls centered on the labeled points. If the k-center problem is NP-hard [Cook et al., 1998], a greedy algorithm provides a solution $\boldsymbol{x}^*_{1:b}$ with $\delta_{greedy} = \max_{\boldsymbol{x}_i \in U_t} \min_{\boldsymbol{x}_j \in \boldsymbol{x}^*_{1:b} \cup L_t} \Delta(\boldsymbol{x}_i, \boldsymbol{x}_j)$ such that $\delta_{greedy} \leq 2 \times \delta_{opt}$, where δ_{opt} is the maximum distance for the optimal solution. The principle of the greedy algorithm is to iteratively select data samples such that they maximize the minimum distance between unlabeled and labeled points:

$$\forall k \in \{1, \ldots, b\}, \boldsymbol{x}^*_k = \underset{\boldsymbol{x}_i \in U_t \setminus \{\boldsymbol{x}^*_1, \ldots, \boldsymbol{x}^*_{k-1}\}}{\arg\max} \; \min_{\boldsymbol{x}_j \in L_t \cup \{\boldsymbol{x}^*_1, \ldots, \boldsymbol{x}^*_{k-1}\}} \Delta(\boldsymbol{x}_i, \boldsymbol{x}_j) \tag{4.9}$$

The result of the greedy algorithm is used as an initialization to perform the search of a better solution. For a given radius δ, [Sener and Savarese, 2018] formulate a mixed integer program $MIP(L_t, U_t, b, \delta, n_{outliers})$ that verifies if the following core-set problem is feasible, where $n_{outliers}$ denote the number of data points that can be removed from the unlabeled pool:

$$\min_{\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_b\} \subset U_t} \; \max_{\boldsymbol{x}_i \in X_u \setminus \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_b\}} \; \min_{\boldsymbol{x}_j \in \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_b\} \cup L_t} \Delta(\boldsymbol{x}_i, \boldsymbol{x}_j) \leq \delta \tag{4.10}$$

This is much easier to perform than solving the k-center problem itself. Starting from the upper bound given by the greedy solution, smaller and smaller bounds are iteratively queried, as described in Algorithm 2. Meanwhile, the weakness of the k-center algorithm, that is prone to select outliers, is tackled by allowing the algorithm to ignore at most a given budget of data points.

The Core-set approach is motivated by the fact that small changes in the feature space should induce small changes in the classifier decision. Therefore, having a training data set that spans over the whole data set should allow the classifier to have a good knowledge of the data. As far as the number of variables in the mixed integer program proportionally grows with the size of the unlabeled pool, and that as many distances as the number of labeled points times the number of unlabeled points are computed, the Core-set technique applied on large data sets is too time-consuming to be used in practice, though selecting points within a subset of the unlabeled set could circumvent the problem at the expense of performance.

Hierarchical sampling The idea behind the technique introduced by [Dasgupta and Hsu, 2008] is to query points so that there are enough training samples in each data cluster to

Algorithm 2: MIP routine of [Sener and Savarese, 2018] to refine the greedy solution

Data: Labeled and unlabeled sets, L_t and U_t, at step t;
Input: Budget b of points to query and budget $n_{outliers}$ of outliers;
Distance δ_{greedy} associated to the greedy solution ;
Initialize: $\delta_{sup} \leftarrow \delta_{greedy}$;
$\delta_{inf} \leftarrow \delta_{greedy}/2$;
A small constant ϵ
while $\delta_{sup} - \delta_{inf} \geq \epsilon$ **do**
  if $MIP(L_t, U_t, b, \frac{\delta_{inf}+\delta_{sup}}{2}, n_{outliers})$ *is feasible* **then**
    $\delta_{sup} \leftarrow \frac{\delta_{inf}+\delta_{sup}}{2}$
  end
  else
    $\delta_{inf} \leftarrow \frac{\delta_{inf}+\delta_{sup}}{2}$
  end
end
Result:

make the classifier confident on the class of every cluster. A hierarchical segmentation $\mathcal{T}$ is computed, starting from a segmentation where each pixel is a region, to a segmentation where all pixels are included in one region, by iteratively merging regions based on a similarity criterion. This hierarchical segmentation can be described as a tree, where the root is the region containing all pixels and the leaves are each single pixel. Then, the hierarchical segmentation $\mathcal{T}$ is pruned (branches are kept or cut), yielding a pruning (*i.e.* a new tree) $\mathcal{P}(\mathcal{T})$. Starting from the root, nodes are kept or removed (by cutting branches) whether the test error prediction is reduced and that nodes are said to be admissible. For a given cluster v classified as its majority label l, the test error ϵ_v is formulated as the portion of misclassified samples. It is estimated by querying and labeling points from the clusters:

$$\epsilon_v = 1 - \max_l p_{v,l} \tag{4.11}$$

where $p_{v,l}$ is the portion of the class l in the cluster v. The confidence interval associated with the class probabilities at step t is $[p_{v,l}^{LB}(t),\ p_{v,l}^{UB}(t)]$ where:

$$p_{v,l}^{LB}(t) = max(p_{v,l}(t) - \Delta_{v,l}(t), 0) \tag{4.12}$$

$$p_{v,l}^{UB}(t) = min(p_{v,l}(t) + \Delta_{v,l}(t), 1) \tag{4.13}$$

$$\Delta_{v,l}(t) \approx \frac{1}{n_v(t)} + \sqrt{\frac{p_{v,l}(t)(1 - p_{v,l}(t))}{n_v(t)}} \tag{4.14}$$

with $n_v(t)$ the number of queried points from cluster v. The higher the number of queried points, the higher the confidence in the class of the cluster. The closer $p_{v,l}$ from 0.5, the lower the confidence. When the confidence is high for a cluster and a label, they are said to be admissible.

The idea is that labeling pixels from some clusters would likely be more informative than labeling pixels from other clusters. For instance, querying pixels from clusters that contain already a lot of samples from the same class will probably not be very informative as it will probably not reduce the average test error. From this perspective, the hierarchical sampling technique could be viewed as a performance-based heuristic. If we consider that the test error ϵ_v^{t+1} for a cluster v at step $t+1$ is a random variable that depends on the labeled set L_t, on the pool U_t, on the associated hierarchical segmentation $\mathcal{T}$ and on random variables $y_{1:b}$

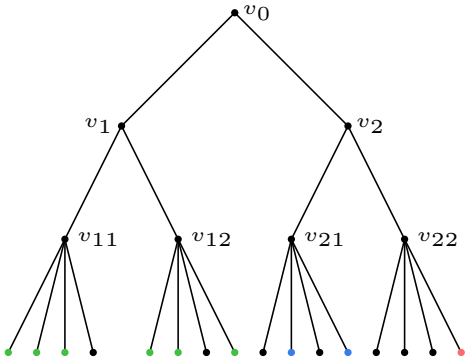


Figure 4.5: Hierarchical segmentation illustrated as a tree. The dots at the bottom of the tree, *i.e.* the leaves, represent every pixels in the data set. The color of dots represent their semantic labels. Black dots are unlabeled data. Because most of the labeled pixels included in node v_1 belong to the same class, the branches between v_1 and v_{11} as well as v_{12} will be cut and querying pixels from v_1 will be unlikely. In contrast, querying pixels from v_2 will be likely because we have a low confidence in its majority class. After querying enough pixels, if node v_{21} mainly contains blue pixels and node v_{22} mainly contains red pixels as we can expect, then nodes v_{21} and v_{22} will be kept in the tree.

which are the classes of the candidate data samples $\boldsymbol{x}_{1:b}$, then the acquisition function of the hierarchical sampling technique could be viewed as follows:

$$a_{hierarchical}(x_{1:b}; y_{1:b}, L_t, U_t, \mathcal{T}) = - \sum_{v \in \mathcal{P}(\mathcal{T})} \mathbb{E}_{\epsilon_v^{t+1} \sim p(\epsilon_v^{t+1} | y_{1:b}, L_t, U_t, \mathcal{T})} \left[\epsilon_v^{t+1} \right] \tag{4.15}$$

Although, the probability distribution over the test error is not known. In practice, points are randomly queried from a cluster v with the probability $w_v(1 - p^{UB}_{v,L(v)}(t))$, where w_v is the portion of points of cluster v in the whole data set. The term $1 - p^{UB}_{v,L(v)}(t)$ reduces the sampling rate in clusters that are very pure, *i.e.* that mostly contain observations with the same label. The pruning $\mathcal{P}(\mathcal{T})$, *i.e.* a clustering from the hierarchical segmentation, is obtained by descending the tree from the root to the leaves as long as the clusters are admissible and the estimated error is reduced, *i.e.* the error of labeling the cluster v by its majority label is greater than the average of the errors of its children.

All in all, the hierarchical strategy is almost like random sampling aside it stops querying points that belong to already well-known clusters. It is worth noting that it relies on the segmentation, that itself depends on the learned feature space.

VAAL A variational auto-encoder projects labeled and unlabeled data points on a joint representation space denoted as $\mathcal{Z}$. A discriminator $D : \mathcal{Z} \longrightarrow [0, 1]$ predicts the probability that a sample is unlabeled: it spots the unlabeled points $x_U \in U_t$ from labeled ones $x_L \in L_t$. The key idea of VAAL is that the VAE and the discriminator are trained in an adversarial fashion. The VAE is optimized in order to both reconstruct the data and to fool the discriminator, thus looking for common features among the labeled and unlabeled points. In contrast, the discriminator seeks points that are unlikely to come from the labeled data set:

$$a_{vaal}(x_{1:b}) = \sum_{k=1}^{b} a_{vaal}(x_k) = \sum_{k=1}^{b} D(x_k) \tag{4.16}$$

The VAE is optimized to maximize a lower bound $\mathcal{ELBO}$ of the log-marginal likelihood of the data and to fool the discriminator, *i.e.*. maximize the log-likelihood for labeled and unlabeled data points to be predicted as labeled ones:

$$\mathcal{L}_{VAE} = \lambda \mathcal{ELBO} + (1 - \lambda) \mathcal{L}^{adv} \tag{4.17}$$

where λ is a hyperparameter and $\mathcal{L}^{adv}$ is defined as follows:

$$\begin{aligned}\mathcal{L}^{adv}(\theta,\phi) = &- \mathbb{E}_{x_L\sim\hat{p}_{labeled-data}(x_L)}[\log\, D(q_\phi(z_L|x_L))] \\ &- \mathbb{E}_{x_U\sim\hat{p}_{unlabeled-data}(x_U)}[\log\, D(q_\phi(z_U|x_U))]\end{aligned} \tag{4.18}$$

where q_ϕ is the encoding part of the VAE. The discriminator, which is the query system, is optimized by minimizing a standard cross-entropy loss. As far as the discriminator is obtained by a mini-batch optimization, VAAL scales well to large data sets.

LAL [Konyushkova et al., 2017] introduced a new active learning paradigm where a meta-model learns how the classifier itself learns. Given the state of the classifier and some candidate data point to query, a regressor learns to predict the test error reduction induced by labeling the candidate data point. The classifier used in [Konyushkova et al., 2017] is specifically a random forest classifier $f : \mathcal{X} \longrightarrow [0,1]^{C\times N}$ where C is the number of classes and N is the number of trees. The state of the model is described with the following features: the proportion of labels in each class, the out-of-bag score (a validation metric), the standard deviation of the feature importances (feature importance is a vector of scalars that sum to one and gives for each feature how influent it is for prediction), the forest predictions variance, the average depth of the trees in the forest and the size of the labeled data set. As for the candidate data points, they are characterized by the mean and the standard deviation of the classifier output. Although the original paper only considers one versus one classification, the features can be easily enhanced for mutli-class problems. Then, for a data point $\boldsymbol{x}$ and a regressor f, the features are concatenated in a feature vector $\xi(x, f)$, which are the input to a regressor $R : \xi \longrightarrow \mathbb{R}$ that predicts the test error reduction. The acquisition function is defined as follows:

$$a_{lal}(\boldsymbol{x}_{1:b}, f) = \sum_{k=1}^{b} a_{lal}(\boldsymbol{x}_k, f) = \sum_{k=1}^{b} R(\xi(\boldsymbol{x}_k, f)) \tag{4.19}$$

In order to train the regressor, the training data set is randomly split into τ labeled and unlabeled sets at each AL step, as well as into a unique test set. For each split, the test error reduction is estimated for M samples from the unlabeled set added to the labeled one. This process is repeated with Q different random splits of L_t. Hence, $Q \times \tau \times M$ samples are used to train the regressor.

■ **Synthesis** Table 4.2 summarizes the strategies of the benchmarked methods and their underlying assumptions. In the next section, we will highlight how those differences materialize in practice through experiments on toy and real data sets.

3.2 Numerical experiments

The objectives of the numerical experiments are:

1. to assess the capacity of AL techniques to improve the *quality* of the training data set,
2. to assess the capacity of AL techniques to highlight land cover classes missing in the training data set,
3. to identify the strengths and weaknesses of the different AL strategies.

First, we present our experimental plan and then, we thoroughly present the results.

Table 4.2: Synthesis of benchmarked AL methods

Method	Acquisition function	Underlying assumption
Breaking Ties	$-\sum_{k=1}^{b}\left(\max_{i\in(1,\dots,C)} f(\boldsymbol{x}_k)^i - \max_{j,j\neq i} f(\boldsymbol{x}_k)^j\right)$ *Point-wise maximum inter-class uncertainty*	Classifier confidence is correlated to uncertainty
Batch-BALD	$H(y_{1:b}\|\boldsymbol{x}_{1:b},\mathcal{D}_{train}) - \mathbb{E}_{p(\boldsymbol{\theta}\|\mathcal{D}_{train})}[H(y_{1:b}\|\boldsymbol{x}_{1:b},\boldsymbol{\theta},\mathcal{D}_{train})]$ *Batch-wise maximum epistemic uncertainty*	Different model parameters yield different predictions for uncertain samples
VAAL	$\sum_{k=1}^{b} D(\boldsymbol{x}_k)$ *Point-wise minimum discriminator likelihood*	Very different samples from the training data are representative of the whole data
Core-set	$-\max_{\boldsymbol{x}_i\in U_t}\ \min_{\boldsymbol{x}_j\in x_{1:b}\cup L_t} \Delta(\boldsymbol{x}_i,\boldsymbol{x}_j)$ *Batch-wise minimum largest distance*	Close samples in the data space belong to the same class
Hierarchical sampling	$-\sum_{v\in\mathcal{P}(\mathcal{T})} \mathbb{E}_{\epsilon_v^{t+1}\sim p(\epsilon_v^{t+1}\|y_{1:b},L_t,U_t,\mathcal{T})}\left[\epsilon_v^{t+1}\right]$ *Point-wise minimum classification error*	Samples in the same cluster belong to the same class
LAL	$\sum_{k=1}^{b} R(\xi(\boldsymbol{x}_k,f))$ *Point-wise maximum test error reduction*	The most informative samples depend on the state of the classifier

3.2.1 Experimental protocol

The numerical experiments consist in running the AL algorithms described in section 3.1 on various data sets for a given number of iterations and a budget of pixels to query. In order to qualitatively assess the benefits of AL with regard to the *quality* of the training set (objective 1), we computed accuracy metrics on a segmentation task after each AL step. To assess the capacity of AL methods to guide the choice of the nomenclature (objective 2), we intentionally left out some classes from the initial training data sets in some cases. Finally, to identify the advantages and drawbacks of the benchmarked techniques (objective 3), we built a toy data set in order to easily visualize the AL process and kept track of an additional metric: the number of queried pixels per class. In the following, we present in details and justify the choice of the data, the metrics, the hyperparameters and the implementations.

■ Metrics

The metrics used to evaluate the active learning methods both in the machine learning and remote sensing communities are usually accuracy metrics given the number of training samples added through the active learning process only. In our experiments, we kept track of the overall accuracy (OA) and of the mean intersect over union metric (mIoU) defined in chapter 3. To have a fair comparison of the methods and because of its simplicity and short training time, we chose a SVM classifier to make the predictions for each AL steps. The SVM parameters were selected through a grid search over the whole labeled data sets. In a real case scenario, however, those metrics are not sufficient as far as they cannot report:

- whether the query system selects samples that do not belong to any of the initial classes (which can be very likely for remote sensing data),
- the diversity of samples acquired at each active learning step,
- the proportion of selected outliers.

Therefore, we also kept track of the number of added pixels for initial and new classes over the total number of training pixels. This metric should also indicate whether the query system fosters intra-class representativeness, epistemic uncertainty or both, as well as its sensitivity to outliers and its diversity.

■ Hyperparameters

For AL algorithms tied with classifiers (Breaking Ties, BALD, Batch-BALD, Core-set, VAAL), we used the state-of-the-art spectral convolutional neural network from [Hu et al., 2015] as far as it was recently benchmarked by [Audebert et al., 2019] and showed good performances. At each step t, the model is trained from scratch on the current training data set. **At each step, we queried 100 pixels as**:

- Less than 100 pixels would hardly be statistically representative,
- In order to assess how fast AL methods improve the classification performance and to evaluate their capacity to reduce the annotation cost, the lower the number of pixels selected at each step, the better. For instance, it would be very interesting for an operational context to know that only one step querying 100 pixels is enough to reach high performance instead of one step querying 200 pixels,
- For a given budget of pixels to label, having few pixels at each step allows to benefit from an updated measure of the acquisition function.

We ran the experiments for 30 steps on Indian Pines, for 15 steps for Pavia University as accuracy metrics stop improving and for 5 steps on Mauzac considering the complexity and time to label pixels as we shall discuss below. We approximated Bayesian neural networks using consistent MC dropout [Gal and Ghahramani, 2016] with a 0.5 dropout rate. For every experiments, we either kept the default hyperparameters (used in the original papers) or used hyperparameters that yielded reasonable memory and time requirements while ensuring the convergence of neural networks optimizations (see table 4.4 for orders of magnitude for time requirements). Hyperparameters for every experiments are given in table 4.3.

Table 4.3: Hyperparameters of the tested AL methods. Common parameters for the optimization of neural networks are the learning rate, the batch size, the number of epochs, the optimizer (we used Adam [Kingma and Ba, 2015]) and the criterion (we used the cross-entropy loss).

Methods	Hyperparameters
Breaking Ties	Common model and training parameters
Batch-BALD	$k = 30$, the number of MC samples to estimate expectations $m = 100$, the number of samples to estimate the joint entropy Common model and training parameters
VAAL	$\beta = 1$, $\lambda = 0.5$ Number of alternate steps between the VAE and the discriminator for the adversarial training : respectively 2 and 1 Common model and training parameters
Core-set	Proportion of admissible outliers: $1e^{-4}$ Subsampling rate: 1 for Indian Pines and 0.25 for Pavia University Common model and training parameters
Hierarchical sampling	$\beta = 2$ Linkage and affinity of the hierarchical clustering
LAL	$Q = 10$, $M = 10$, $\tau = 5$ Model parameters (number of predictors = 50)

■ Data

In order to illustrate the practical behavior of the compared methods, we built a 2D **toy data set** from a hyperspectral scene. We selected different samples from two classes of the Mauzac data set (cf chapter 3): *Dry Vegetation (1)* and *Tile (2)*. We performed a Principal Component Analysis [Tipping and Bishop, 1999] over the selected samples and kept the first two principal components. On fig. 4.6, data points are plotted according to their two first principal components. Labeled and unlabeled data points from class *(1)* are represented by green circles and dark green squares respectively. Labeled and unlabeled data points

from class *(2)* are represented by pink circles and purple squares respectively. We can note that classes *(1)* and *(2)* have important intra-class variability and inter-class similarities: their samples indeed split into distinct clusters. We labeled samples from one cluster only in order to compare the methods in a challenging initial setting. **Indian Pines** and **Pavia University** data sets allowed to automate the experiments as far as ground truth are provided with the images. Therefore, we could split the ground truths in 5 different disjoint splits of the initial training set, an unlabeled pool from which pixels are queried (and automatically labeled) and a test set on which accuracy metrics were computed.

In order to assess whether AL techniques can explore the data and guide the choice of the nomenclature, we experimented two different settings on the Pavia University data set:

- Setting (1): every classes are represented in the initial training data set,
- Setting (2): classes *Gravel*, *Trees*, *Painted metal sheets* and *Bare soil* are not represented in the initial training data set.

Finally, the **Mauzac** image is much more representative of an operational use case. We labeled an initial ground truth of the Mauzac image with the most present land cover classes: *High vegetation*, *Ground vegetation*, *Dry vegetation*, *Water body*, *Tile* and *Asphalt*. Approximately 3,500 pixels are labeled, which means that there are still more than 800,000 pixels on the image of Mauzac for which we don't know the true classes. Therefore, the pool size of the Mauzac data set is 26 times larger than the one of Pavia University and 120 times larger than the one of Indian Pines. Here we emphasize on the fact that, on benchmark data sets, AL methods are applied on pixels for which we already know the true classes. In contrast, we apply AL methods on every pixels of the Mauzac image, except those in the training data set, which has the following consequences:

- We have to label the pixels ourselves by photo-interpretation and on-field campaigns,
- Some pixels may be mixed pixels,
- Some pixels may not belong to any of the classes in the initial training data set,
- Some pixels may not be relevant of the land cover such as containers, garbages or cars,
- We may not always label pixels with confidence, and we even might have to label some pixels as *Unknown*.

Annotating pixels by photo-interpretation and field campaigns is difficult, time-consuming and expensive. Labeling the pixels for one query (one method and one step) took on average a bit more than one hour. Thus, the annotation for the 7 methods over 5 steps approximately took 40 hours (including a field campaign completed beforehand that took half a day). For those reasons, we could not run more than 5 steps per method on Mauzac. After we ran the AL methods on Mauzac, we have labeled an independent test data set with the initial and some of the new discovered classes in order to compute quantitative metrics. We kept in the test data set classes for which we could label pixels with confidence, and left out the *Unknown* class as well as classes of mix materials, resulting in 16 classes: *High vegetation*, *Ground vegetation*, *Dry vegetation*, *Water body*, *Tile*, *Asphalt*, *Gravel*, *Bare soil*, *Cement*, *Shadow*, *Vegetation shadow*, *Plastic*, *Roads white lines*, *Painted sheet metal*, *Curbstone* and *Swimming pool*.

To sum up, we consider four data sets with varying complexity in our experiments. The toy data set illustrates how the methods behave. Indian Pines is a quite challenging data set despite its small size. As a matter of fact, it has 16 classes of mixed materials, essentially crops, with very high intra-class variabilities and inter-class similarities. Pavia University is larger but is more homogeneous with a smaller spectral domain. State-of-the-art classifiers indeed achieve much better accuracies on Pavia University than on Indian Pines [Audebert

et al., 2019]. Finally, Mauzac is a complex image where all pixels are considered. It illustrates the problems one can encounter using active learning in a life-like scenario.

■ Preprocessing mixed pixels

In a real use case where the entire image is considered, we suggest the following preprocessing routine to deal with mixed pixels that are not of interest:

- Apply Sobel filters [Sobel, 2014] on the hyperspectral image averaged over every bands,
- Compute a mask by applying a threshold to the Sobel edge map,
- Apply a closing operation [Vincent, 1993] on the obtained mask,
- Apply the mask to the hyperspectral image.

Setting the threshold and the structuring element of the morphological operation correctly allows to remove most edges and small objects whose size is approximately smaller than $2m \times 2m$, such as cars that are not relevant to the land cover.

■ Preprocessing large images

Some query systems such as Core-set and Hierarchical sampling do not scale on large data sets: their are impractical due to long computation times. A very simple way to circumvent this issue is to perform the AL step on a random subset of the pool, which we denote as subsampling. On the Mauzac data set however, the subsample rate would have to be so small to have reasonable computation requirements that it would considerably reduce the efficiency of AL. Therefore, we suggest a data preprocessing routine based on superpixels segmentation. The AL query is performed on the superpixels rather than on the pixels themselves. The preprocessing technique and its impact on the AL performance will be studied in section 4. In our experiments, we used subsampling for Pavia University and superpixel-based segmentation for Mauzac for the Core-set and Hierarchical sampling techniques.

■ Implementations

AL algorithms were implemented with Python using the paper original code when possible or re-implementations. Algorithms are brought under the same framework and other AL strategies can easily be added in our toolbox[1]. We notably used the Baal library [Atighehchian et al., 2019] for BALD and Batch-BALD. Neural networks were implemented with PyTorch, Core-set MIP problem was solved with OR-Tools while the hierarchical segmentation and the LAL random forest were implemented with Sklearn. Experiments are conducted on one CPU Intel Cascade Lake CLX-6230 20c 2,1GHz and 192GB memory and one GPU Nvidia RTX6000.

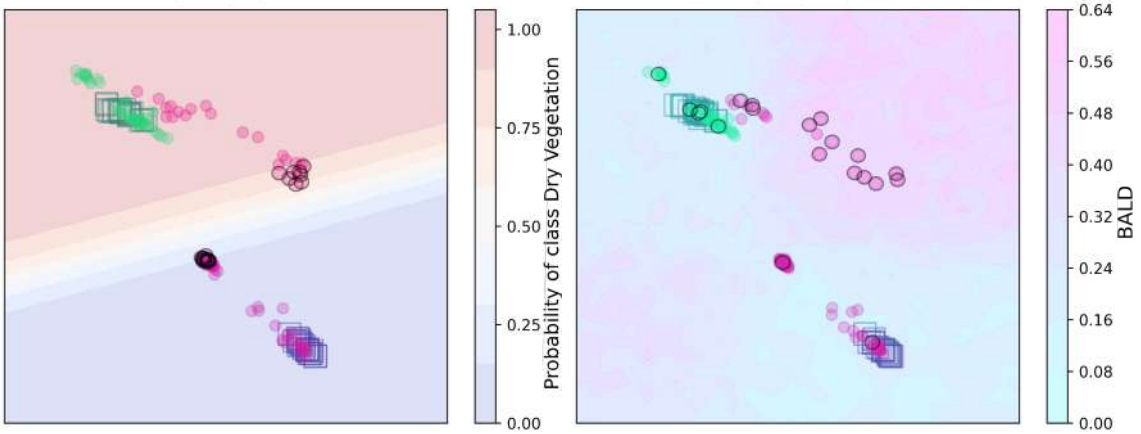
3.2.2 Results

■ Toy data set

Experiments on the toy data set were conducted for illustrative purpose only. We hope that the following observations provide intuition on how the methods work as well as some expected pros and cons. Fig. 4.6 shows the first AL step, for which 20 samples were selected.

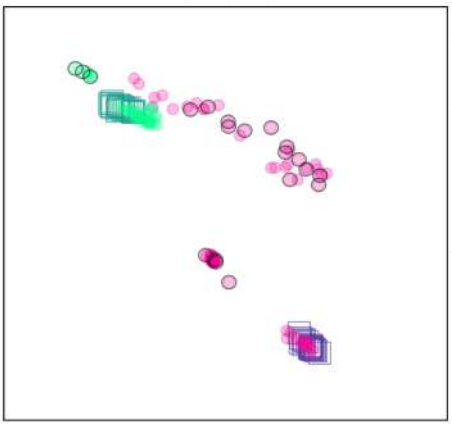
Breaking Ties, by selecting points near the classifier decision boundary, might require many steps before yielding a training data set representative of every clusters.

[1] https://github.com/Romain3Ch216/AL4EO

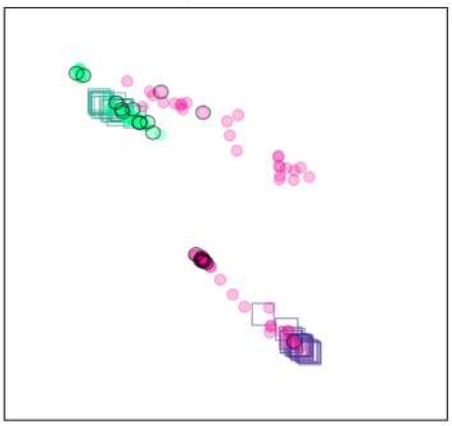


(a) Breaking Ties

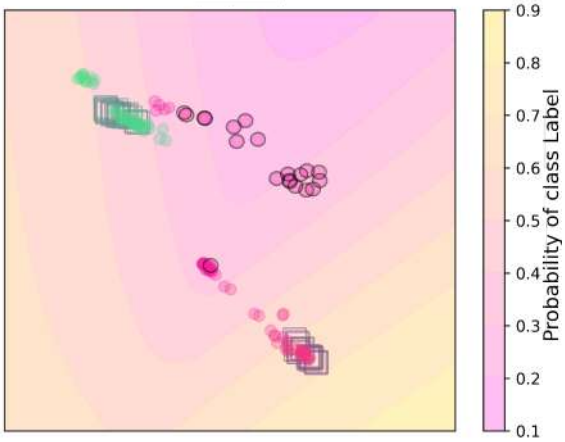
(b) Batch Bald



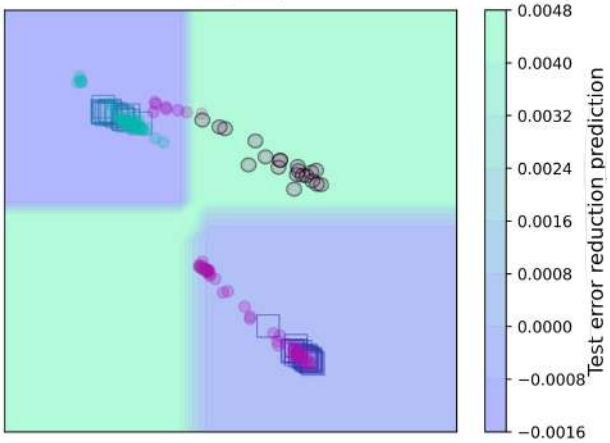
(c) Core-set



(d) Hierarchical sampling



(e) VAAL



(f) LAL

Figure 4.6: Illustration of the query systems on the toy data set. Labeled and unlabeled data points from class *(1)* are represented by green circles and dark green squares respectively. Labeled and unlabeled data points from class *(2)* are represented by pink circles and purple squares respectively. At first, they are located in the bottom-right and in the top-left clusters. Queried points at step 1 are circled by black disks. *(a)* Breaking Ties: the background color represents the probability of a sample to belong to class *Dry Vegetation (1)* according to the trained classifier. *(b)* Batch-BALD: the background color represents the BALD score. *(c)* Core-set. *(d)* Hierarchical sampling. *(e)* VAAL: the background color represents the probability (given by the discriminator) of samples to come from the labeled data set. *(f)*: the background color represents the test error reduction predicted by LAL regressor.

Batch-BALD mainly focuses on the region of the data space without any training samples where the BALD score is high. As a matter of fact, the BALD metric measures the disagreement between confident predictions with different draws of the model parameters. In regions without training samples, the classifier mapping from inputs to outputs can take any value, leading to a high disagreement. Batch-BALD also selects points belonging to different clusters, bringing more diversity to the labeled data set. If points can be individually informative on the right model parameters, they can also be redundant. Defining Batch-BALD as the mutual information between the points predictions and the model parameters allows to prevent such redundancy.

Core-set directly explores all unknown regions in order to reduce the maximum distance between the closest labeled and unlabeled points. We see that, in this way, Core-set is likely to select a batch of points that exhibits at the same time high inter-class uncertainty and high epistemic uncertainty.

Hierarchical sampling, in the first instance, is more likely to query points in dense clusters. Like Core-set, it has the advantage to select points in different regions of the data space.

VAAL selects unlabeled data points that are far from labeled data points (having a low probability to come from the labeled data set). A weakness highlighted on the toy data set is that it does not consider the known classes of the labeled samples as well as the classes predictions. In other words, the information given by the annotations is not used by VAAL. Thus, if the VAE is too good to find common representations of samples coming from different classes, the discriminator might miss very informative samples.

Finally, it seems that in the present case, **LAL** mainly exploits the variance predictions to target points that lie in regions where the random forest has little experience.

■ Indian Pines

Overall accuracy (fig. 4.8a). Breaking Ties and random sampling achieved the best overall accuracy at every steps, from 0.61 to 0.70, with a quick increase during the first ten steps. Core-set and Hierarchical sampling obtained almost as good overall accuracies (approximately 1% less). BALD, VAAL and LAL converged to overall accuracies higher than 0.67 but with a very slow increase. Batch-BALD was significantly worse than others (more than a 5% gap).

Mean Intersect over Union (fig. 4.8b). In order of decreasing metrics, Core-set, Breaking Ties, BALD and Hierarchical sampling achieved better performances than random sampling. This result makes sense because random sampling selected few points from little populated classes, yielding poor IoU scores for those classes. LAL and VAAL performed worse than random sampling at the first ten steps then reached comparable metrics. Batch-BALD performed significantly worse.

Proportions of queried pixels (fig. 4.7). Figure 4.7 shows from which classes the queried pixels belonged to, after steps **5**, **10** and **15**. Given the imbalance of the data set and in order to have an easier interpretation of the results, the proportion of added pixels is shown per class. Random sampling virtually queried the same proportion of pixels in each class, as one would expect. Breaking Ties selected notably less pixels (in proportion) for classes *Grass-trees*, *Hay-windrowed*, *Wheat*, *Woods* and *Stone-Steel-Towers*. Here it can be noted that *Stone-Steel-Towers* is the only class that is not vegetation, which makes it easy to differentiate from other classes. Batch-BALD selected a huge portion of pixels (up to 70%) for classes *Grass-pasture*, *Woods* and *Building-Grass-Trees-Drives*. BALD selected more than half of pixels in classes *Alfalfa*, *Grass-pasture-mowed*, *Hay-windrowed* and *Stone-Steel-Towers*. Here we draw the reader attention to the fact that those classes are present in

low proportions (respectively 0.40%, 0.3%, 4.7%, 0.2% and 0.9%). VAAL also selected large portions of *Alfalfa*, *Grass-pasture-mowed*, *Hay-windrowed* and *Stone-Steel-Towers*. Core-set notably selected pixels from *Alfalfa*, *Corn*, *Grass-pasture-mowed*, *Oats* and *Building-Grass-Trees-Drives* that are also present in rather small proportions. Hierarchical sampling selected pixels in approximately equal proportions in each class. Finally, LAL selected large proporions of *Alfalfa*, *Grass-pasture-mowed* and *Hay-windrowed*.

■ Pavia University

⋆ Setting (1)

Overall accuracy (fig. 4.9a). First, we notice that the overall accuracy is very high from the start and that the methods show small deviations. Breaking Ties, Hierarchical sampling and random sampling achieved approximately the same results. Core-set (with subsampling) slowly increased but reached the same performance than previous methods after ten steps. In order of decreasing metrics, BALD, VAAL, LAL and Batch-BALD performed significantly worse (more than 2% gap).

Mean Intersect over Union (fig. 4.9b). Breaking Ties and Core-set (with subsampling) demonstrated significant gains against random sampling and Hierarchical sampling after a few steps. In order of decreasing metrics, BALD, LAL, VAAL and Batch-BALD performed worse.

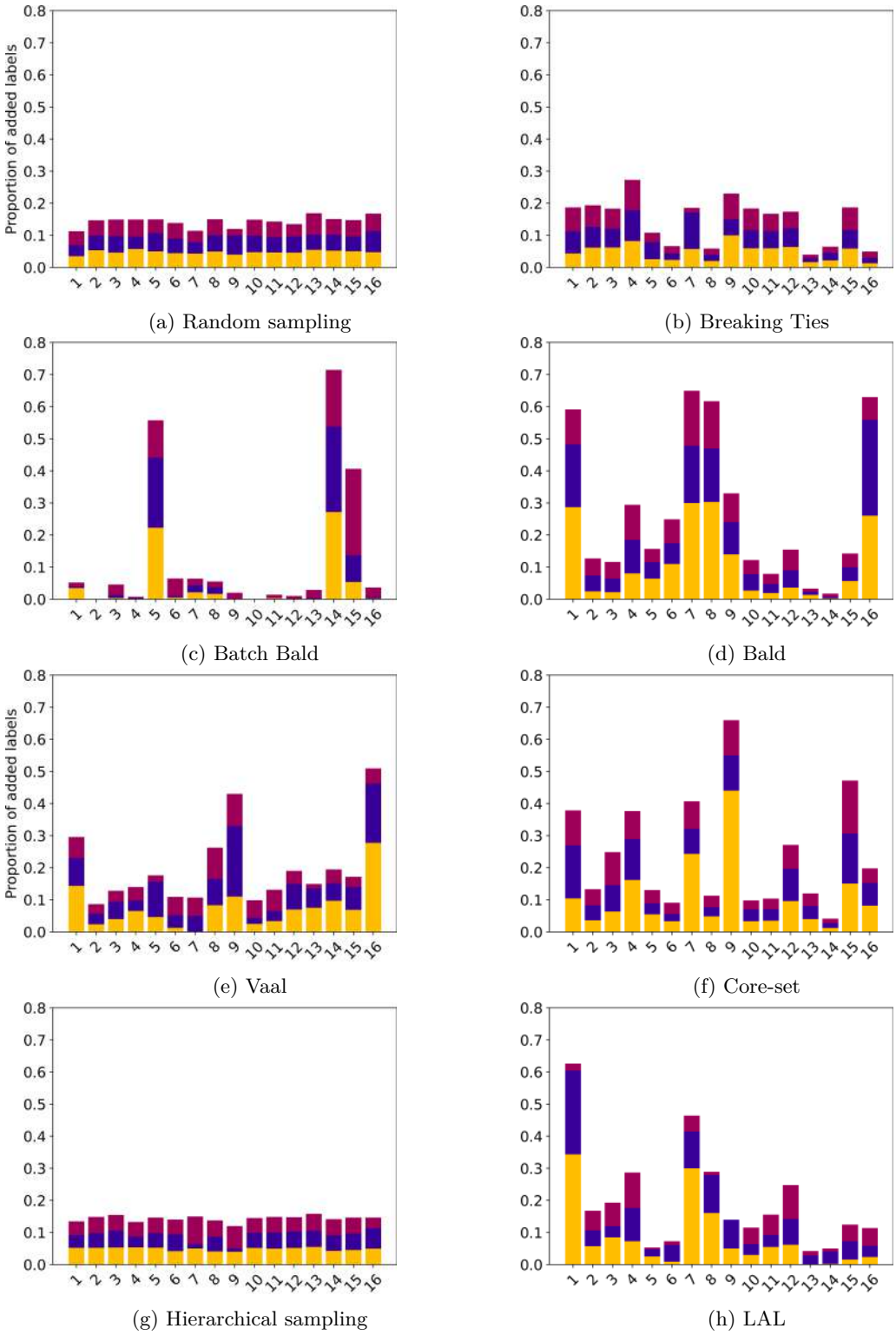
Proportions of queried pixels (fig. 4.11). Random sampling homogeneously queried pixels in each class. Breaking Ties queried a large proportion of *Gravel* but no pixels of *Painted Metal Sheet* and *Shadows*, that can be both very easily discriminated from other classes, as we can see in the presentation of the data set in chapter 3. On the contrary, Batch-BALD queried half of *Painted Metal Sheet* and *Shadows* but almost no pixels from other classes. BALD selected large portions of *Painted Metal Sheet*, *Bare Soil* and *Self-blocking Bricks*. VAAL selected large portions of *Trees*, *Painted Metal Sheet* and *Shadows*. Samples selected by Core-set are more spread out over the different classes. It queried few portions of *Meadows*, *Trees* and barely no pixels from *Painted Metal Sheet* and *Shadows*. Hierarchical sampling had the same behavior than random sampling. LAL took pixels from *Meadows*, *Trees*, *Painted Metal Sheet*, *Bare Soil* and *Shadows* in little proportions.

⋆ Setting (2)

Overall accuracy (fig. 4.10a). Core-set (with subsampling) followed the accuracy of random sampling for the first six steps. Then, it reached a better metric (between 1% and 2% higher). Breaking Ties, Hierarchical sampling and VAAL started from a much lower accuracy and slowly reached the performance of random sampling. BALD started as high as Core-set but did not lead to significant improvements afterwards. Batch-BALD and LAL performed worse.

Mean Intersect over Union (fig. 4.10b). Core-set (with subsampling) demonstrated a significant gain against all other methods from the start to the end. Breaking Ties was also better than random sampling. Hierarchical sampling and VAAL showed the same results than random sampling. In decreasing order of accuracy, BALD, LAL and Batch-BALD performed worse.

Proportions of queried pixels (fig. 4.11). Random sampling homogeneously queried pixels in each class. Breaking Ties behaved like in setting (1), except that it queried pixels from *Painted Metal Sheet*, mostly at first step as it was not labeled in the initial data set. It selected the same portion of *Trees* but only at steps 2 and 3. Batch-BALD behaved quite the same than in setting (1). BALD showed little differences between the two settings, as well as



(a) Random sampling

(b) Breaking Ties

(c) Batch Bald

(d) Bald

(e) Vaal

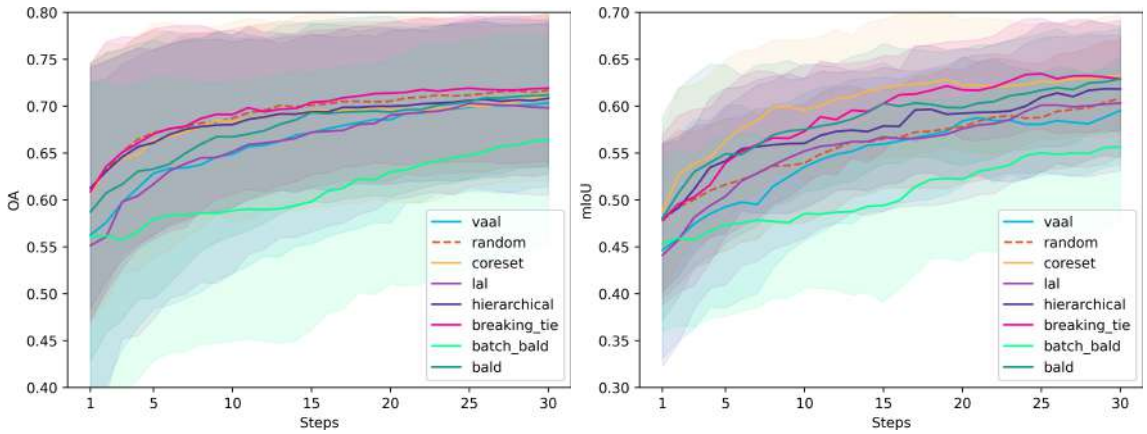
(f) Core-set

(g) Hierarchical sampling

(h) LAL

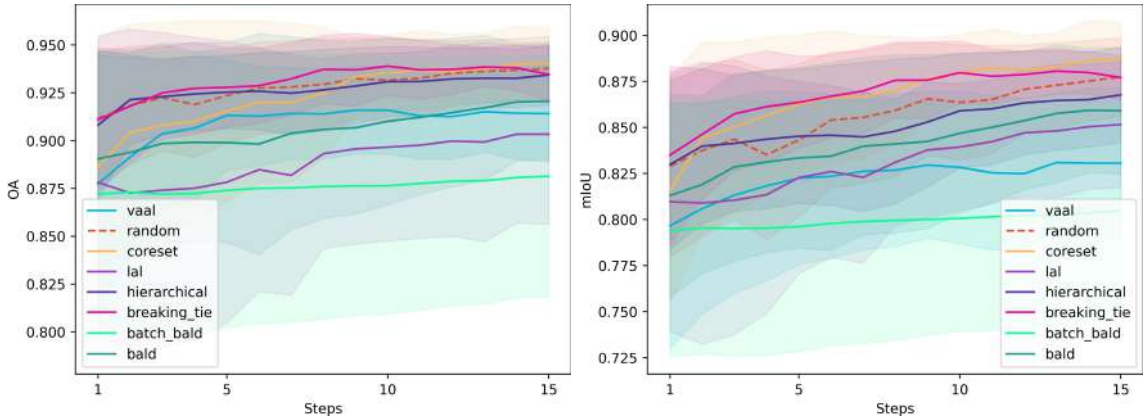
Class id	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Name	*Alfalfa*	*Corn-notill*	*Corn-mintill*	*Corn*	*Grass-pasture*	*Grass-trees*	*Grass-pasture-mowed*	*Hay-windrowed*	*Oats*	*Soybean-notill*	*Soybean-mintill*	*Soybean-clean*	*Wheat*	*Woods*	*Buildings-Grass-Trees-Drives*	*Stone-Steel-Towers*
Proportion (%) of samples in the initial training data set	0.40	14	8.1	2.3	4.7	7.1	0.3	4.7	0.2	9.5	24	5.8	2.0	12	3.8	0.90

Figure 4.7: For each class, proportion of queried pixels after steps 5, 10 and 15 on Indian Pines. Class labels and their proportions in the Indian Pines training data set are given in the table above.



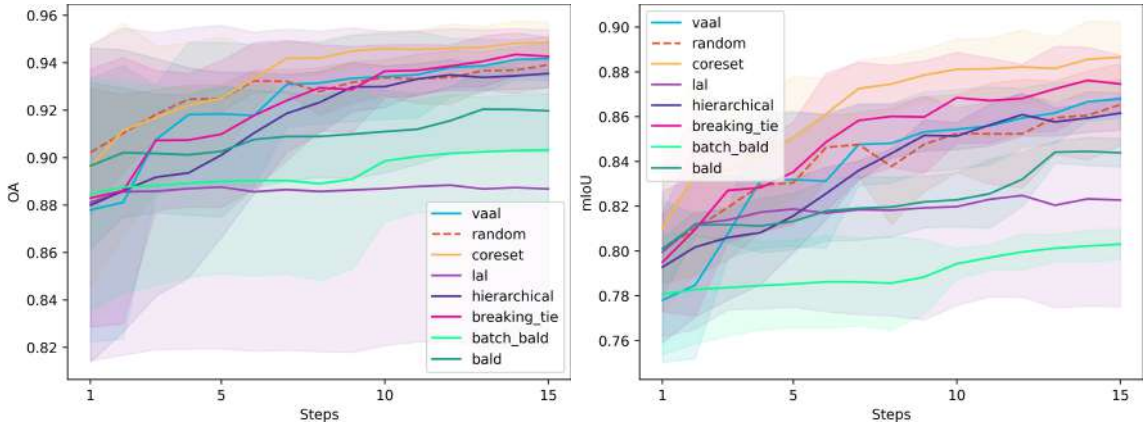
(a) (b)

Figure 4.8: Accuracy metrics (mean and standard deviation over 5 runs) over the first 30 steps of the AL process on **Indian Pines**. (a): overall accuracy. (b): mean intersect over union.



(a) (b)

Figure 4.9: Accuracy metrics (mean and standard deviation over 5 runs) over the first 15 steps of the AL process on **Pavia University** in setting (1).
(a): overall accuracy. (b): mean intersect over union.



(a) (b)

Figure 4.10: Accuracy metrics (mean and standard deviation over 5 runs) over the first 15 steps of the AL process on **Pavia University** in setting (2). (a): overall accuracy. (b): mean intersect over union. Note that Core-set is used with a subsampling of 25% of the unlabeled pool.

VAAL, Core-set and Hierarchical sampling. LAL, compared to setting (1), queried no pixels from *Trees* but selected pixels from *Bare Soil*.

■ **Mauzac**

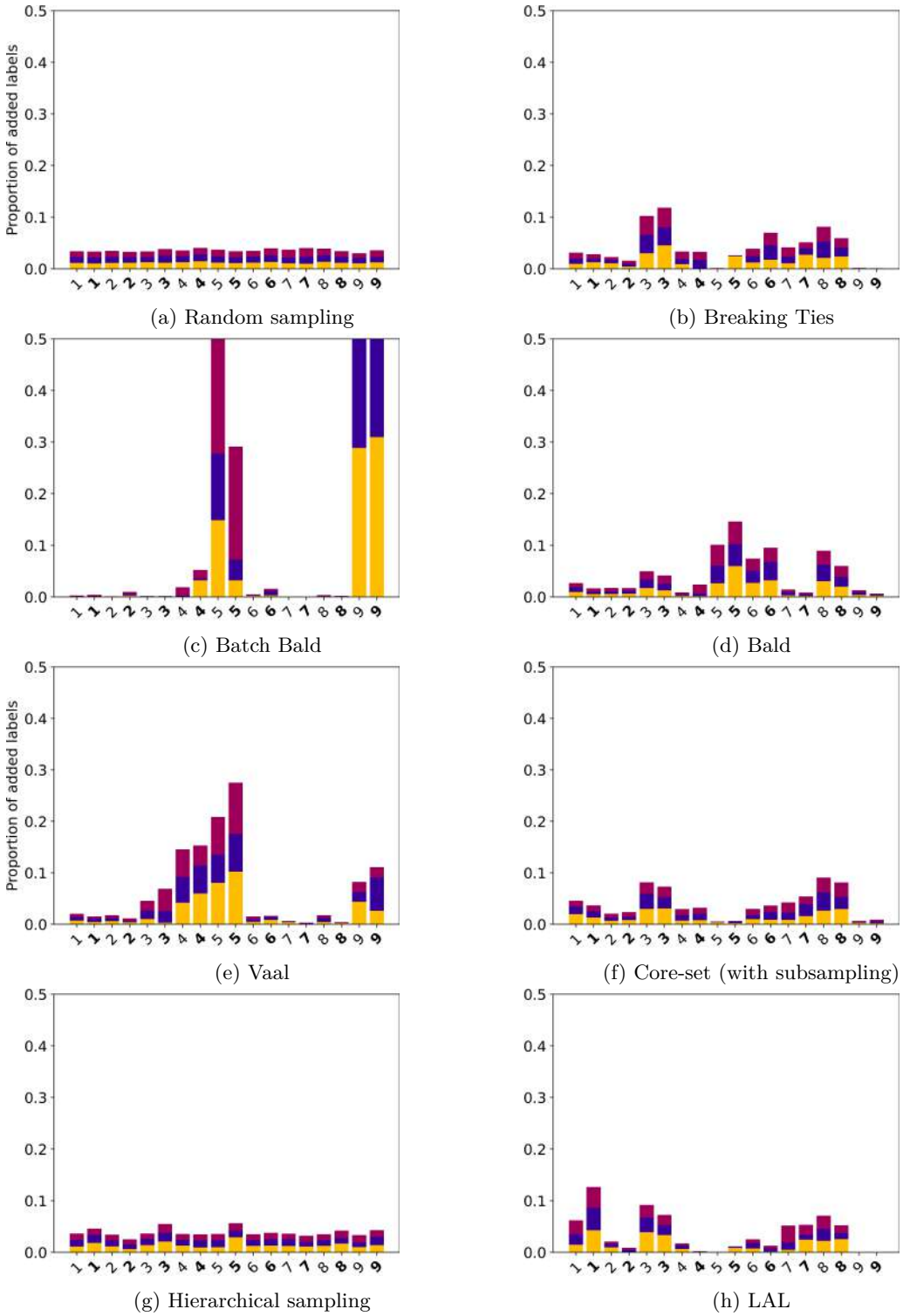
Overall accuracy (fig. 4.12a). Core-set and Hierarchical sampling, applied with our k-means preprocessing method, as well as Breaking Ties and BALD significantly outperformed random sampling (more than 20% higher overall accuracy). Core-set demonstrated a quick increase but stagnated after step 3. On the contrary, Hierarchical sampling and BALD showed little improvements at first steps but a large increase at step 5. VAAL, although better than random sampling showed no improvement in the course of the process. LAL reached much worse accuracy.

Mean Intersect over Union (fig. 4.12b). Hierarchical sampling, followed by BALD, reached the best mIoU at step 5 despite poor performances before step 4. Core-set and Breaking Ties demonstrated a faster increase of mIoU but a slowdown after step 3. LAL and VAAL did not show gains against random sampling.

Proportions of queried pixels (fig. 4.13). At first step, Breaking Ties mainly queried pixels that come from vegetation classes, or a mix of vegetation with other materials such as gravels. Thoses classes exhibit high inter-class similarities. It also queried samples from previously unlabeled classes:

- some mixtures of classes such as *Soil - Vegetation - Gravel* that can be found in gardens or on the roadside. Their spectral signatures lie, by definition, at the frontiers of pure classes,
- *Bare Soil* whose samples might be confused with *Dry Vegetation* that is in reality a mixture of dry vegetation and soil or with *Tile* that is baked clay,
- *Pavement (Cement)* that is spectrally close to *Asphalt*,
- Pixels we were unable to recognise that we labeled as *Unknown*.

At steps 2 and 3, other interesting classes were discovered such as *Gravel* and *Plastic*. The query system especially focused on *Pavement (Cement)* that shares common features with *Gravel* and *Asphalt*. At steps 4 and 5, Breaking Ties kept querying vegetation pixels, *Gravel*, *Asphalt* and *Roads White Lines* that are close to *Gravel* and *Asphalt*, as well as mixed materials such as *Asphalt - Gravel*. BALD discovered more classes than Breaking Ties, but more importantly in fewer steps, such as *Painted Sheet Metal* or *Roads White Line* as well as much more unknown materials. What is worth noting is that it also queried many pixels from vegetation classes. At steps 4 and 5, it queried many pixels from unknown classes and from *Bare Soil*, as well as few pixels from many other classes. VAAL also discovered interesting classes such as *Swimming Pool* or *Curbstone*. However, the discriminator focused mostly on the class *Soil - Vegetation - Water* from step 2 to step 3 and on *Ground Vegetation* at step 4, leaving out other classes. The class *Soil - Vegetation - Water* corresponds to aquatic vegetation on the surface of water bodies or to very shallow water areas where the spectral signature of soils and water is mixed. Core-set virtually queried pixels from each class at every steps (including new classes from step 1). We see that the number of pixels queried from unknown classes increased over the steps. Hierarchical sampling also queried pixels from each class at almost every steps. Compared to Core-set, it discovered less classes, especially less classes of mixed materials. LAL essentially queried only vegetation samples, missing many relevant samples of the pool. Finally, random sampling mostly queried pixels from classes that are highly represented in the image, that is the classes that were initially present in the training data set.



(a) Random sampling

(b) Breaking Ties

(c) Batch Bald

(d) Bald

(e) Vaal

(f) Core-set (with subsampling)

(g) Hierarchical sampling

(h) LAL

Class id		1	2	3	4	5	6	7	8	9
Name		Asphalt	Meadows	Gravel	Trees	Painted Metal Sheet	Bare Soil	Bitumen	Self-Blocking Bricks	Shadows
Proportion (%) of samples in the initial training data set	setting (1)	16	44	4.9	7.2	3.1	12	3.1	8.6	2.2
	setting (2)	16	44	0.0	0.0	0.0	0.0	3.1	8.6	2.2

Figure 4.11: For each class, proportion of queried pixels after **5**, **10** and **15** on Pavia University. Class labels and their proportions in the Pavia University training data set are given on the x-axis. Bolded classes refer to setting (2).

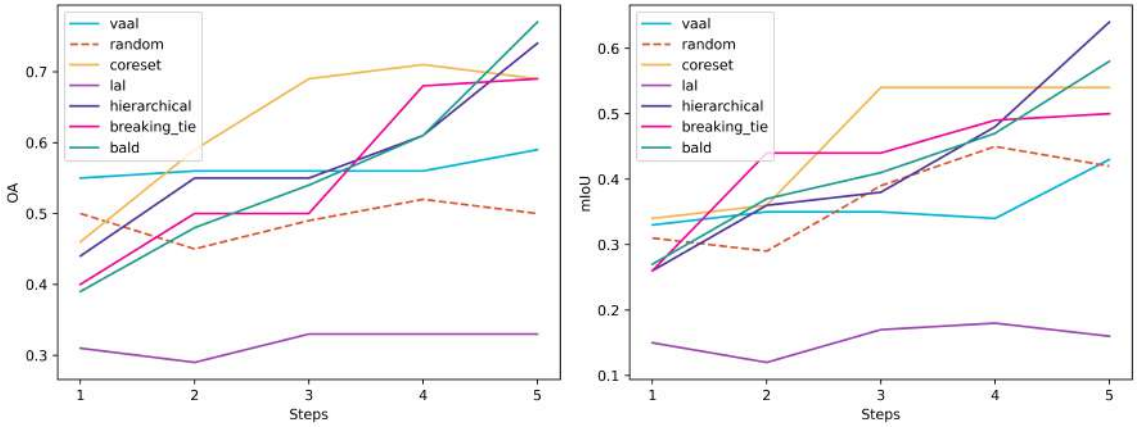


Figure 4.12: Accuracy metrics over the first 5 steps of the AL process on **Mauzac**. (a): overall accuracy. (b): mean intersect over union. Note that Core-set and Hierarchical sampling are applied with our preprocessing method using approximately 10000 clusters.

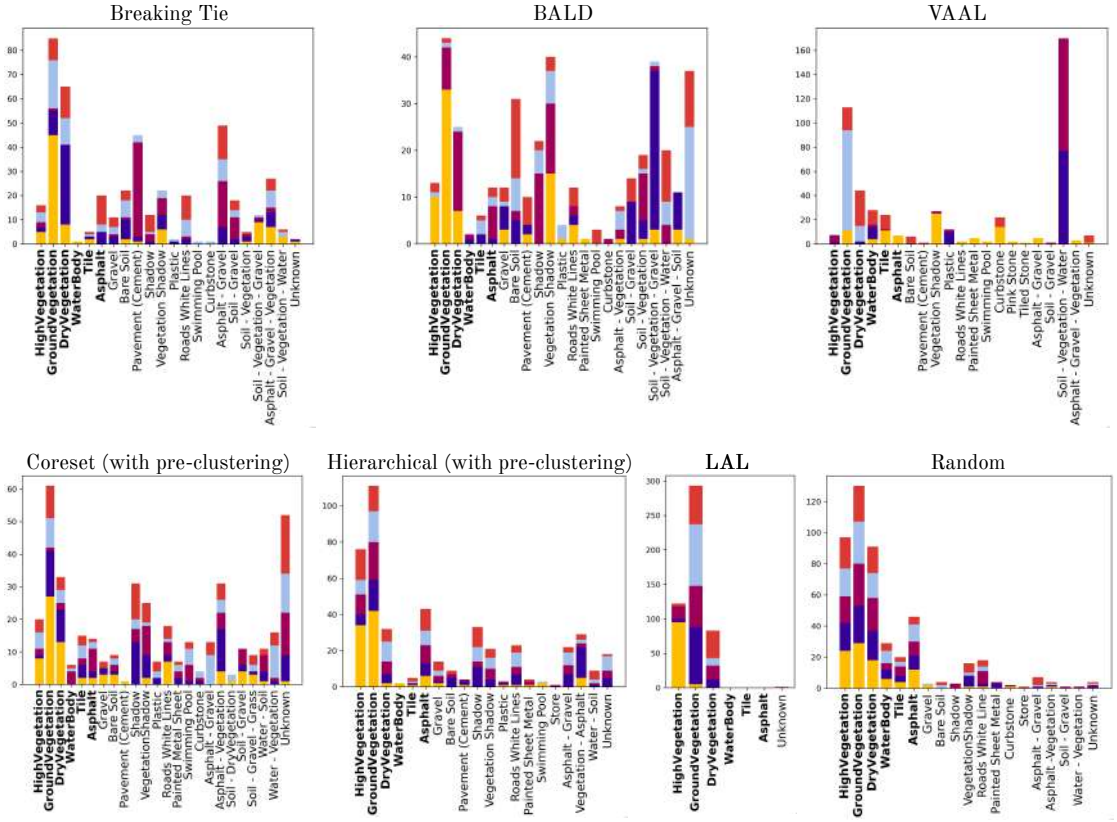


Figure 4.13: For each class, number of queried pixels at steps 1, 2, 3, 4 and 5 on Mauzac. Bolded labels refer to initial classes of the training data set.

Table 4.4: Order of magnitude (in minutes) of sampling time (including training time) with hardware and hyperparameters used in our experiments.

	Indian Pines	**Pavia University**	**Mauzac**
Breaking Ties	1	1	1
BALD	1	1	1
Batch-BALD	1	1 - 10	-
VAAL	1	1	10
Core-set	1	10	10
Hierarchical sampling	1	1	10
LAL	1	1 - 10	1 - 10

3.3 Discussion

Before we discuss the pros and cons of each method, we start with two general remarks. First, the standard deviation of the metrics are considerable, highlighting the fact that the initial training data set and the test set have a high influence on the performances of the machine learning models and the AL algorithms. Second, we should point out that the difference of accuracies between some methods is sometimes quite small and that we shall be careful when drawing conclusions.

Breaking Ties is one of the most competitive method. With a very simple mathematical formulation and low memory and time requirements, it achieved good performances, especially for classes with high inter-class similarities and intra-class variabilities. In contrast to our previous beliefs, it can discover new classes, although not as quickly than BALD or Core-set. Our experiments showed that Breaking Ties only queries necessary samples, it does not seek representativity. In other words, it does not waste time on easily recognizable pixels. The counterpart of this feature, however, is that Breaking Ties can focus on classes that are, by nature, difficult to label and to distinguish, such as healthy vegetation and stressed vegetation. More labeled pixels from those classes will eventually be useless and would not reduce the inter-class uncertainty.

Batch-BALD performed very poorly in our tests. We would expect it to be better than BALD and to acquire more diverse pixels. However, the batch-BALD metric is much more difficult to estimate than BALD. We assume that, in our experiments, the joint entropy is very badly approximated as we only draw 100 samples out of 16^{100} configurations (in the case of Indian Pines which has 16 classes and in the case of 100 pixels queried per step). Increasing the number of draws and decreasing the size of the queried batches might ease the approximation. In pratice though, working with smaller batches would yield a very tedious AL process while increasing the number of draws would require big memory requirements, especially when working on large data sets. Experimenting 2000 draws instead of 100 only has not showed improvements while considerably extending the sampling time. Therefore, we argue that Batch-BALD is not appropriate for remote sensing data.

BALD on the contrary, performed quite well and demonstrated interesting properties in an operational context. From our experiments, it has a good capability to find pixels that belong to little populated classes, or even to classes that were not represented in the initial training data set. Although having a slightly higher computational burden than Breaking Ties, it can be easily used on large data sets.

VAAL showed poor performances compared to random sampling. It seems that it does not suit to data sets with high inter-class similarities. As a matter of fact, the variational autoencoder, in this situation, easily fools the discriminator for unlabeled pixels which are spectrally close to labeled pixels. By focusing on very dissimilar classes (from the most populated classes), it misses crucial information for classification.

Core-set significantly outperformed other methods over the very first steps of AL. It quickly drove up the mIoU score as it directly queries samples far from the labeled samples that span in little-populated clusters and does not focus on a peculiar region of the data space. By tackling active learning as a core-set problem within the prediction space of the classifier, it can be viewed as an inter-class and epistemic uncertainty method all at once. As a matter of fact, the k-center problem leads to samples with high epistemic uncertainty while solving this problem in the output space of the classifier fosters inter-class uncertainty. Samples at the frontier of two classes indeed yield a prediction with high entropy that is far from the low entropic predictions of the labeled samples. It is however quite sensitive to outliers as

showed on the Mauzac experiments. Increasing the proportion of admissible outliers would probably improve its robustness. However, this hyperparameter cannot be as easily tuned as we would usually do with a validation data set. Another downside of Core-set is its memory and time requirements that make subsampling or pre-clustering necessary on large data sets.

Hierarchical sampling demonstrated good results on the most complex data sets Indian Pines and Mauzac, in terms of overall accuracy and mean intersect over union. Compared to uncertainty heuristics, it does not focus on particular classes and showed a high robustness to outliers by querying few pixels from unknown classes.

LAL performed much worse than random sampling. We can wonder whether the features of LAL are relevant in a case of such important inter-class similarities. The increase of the sampling hyperparameter M from 10 to 100 as well as the increase of Q from 10 to 100 did not yield better results while increasing the sampling time by a factor of 10. Besides, looking at the feature importances of LAL regressor, we noted that it gave a great importance to the variance of the probability of classes *High Vegetation* and *Ground Vegetation* on the Mauzac data set which causes this particular focus. More generally, LAL seems to fail in extrapolating on the unlabeled pool. Finally, its high number of hyperparameters as well as its possible long sampling time hinder its practical use.

All in all, Core-set and Breaking Ties consistently stand out from the experiments while BALD demonstrated interesting properties in a life-like scenario. Core-set led to a quick increase of accuracy metrics on every data sets. Along with BALD, it demonstrated great capabilities to quickly discover new classes, which is a valuable feature for complex heterogeneous images. Breaking Ties is particularly relevant to refine the decision boundaries between very similar classes. Therefore, we believe that those techniques may be use in a complementary way in an operational case.

4 Decreasing AL computational requirements

The numerical experiments presented in section 3.2 have shown the benefits of active learning for semantic segmentation of hyperspectral images. However, some methods are limited by their computational burden. On the largest data sets, we indeed had to process the data beforehand to reduce the memory requirements and computation time, which might affect the AL performance. In this section, we present in details the preprocessing techniques and experimentally study their impact on AL efficiency.

4.1 Preprocessing techniques

Random subsampling The simplest and computation-free approach to run computationally demanding AL techniques on large data sets is to take, at each step, a random subset of the pool.

Only slightly more computationally demanding techniques consist in segmenting the pool in superpixels on which the AL is applied, which is related to several region-based AL techniques such as [Cai et al., 2021; Kasarla et al., 2019]. The AL query system selects the most informative superpixels, characterize by radiometric and geometric features such as the spectral average. From the selected superpixels, only a subset of pixels are queried in order to avoid redundancy. We consider two segmentation strategies: k-means clustering [Lloyd, 1982] and SLIC segmentation [Achanta et al., 2012].

(a) k-means (b) SLIC

Figure 4.14: Visualization of superpixel-based preprocessing on the unlabeled pool of the Pavia University image (approximately 30,000 pixels were aggregated into 1000 superpixels). The k-means algorithm results in a segmentation for which some clusters gather one pixel only while some clusters gather many pixels. On the contrary, the SLIC algorithm prioritizes clusters of similar size over the spectral homogeneity.

k-means clustering At each step, the unlabeled pool is segmented with a k-means clustering algorithm, where k denotes the number of superpixels. The clustering is performed in the output space of the classifier. Therefore, the segmentation changes at each AL step.

SLIC segmentation Before the first AL step, the unlabeled pool is segmented with the SLIC (Simple linear iterative clustering) algorithm. The SLIC algorithm is based on the k-means algorithm. k clusters are initialized so that they spatially spread in a homogeneous way on the image. If N is the number of pixels in the image, then the initial size of a superpixel is $S^2 = \frac{N}{k}$. Then, each pixel is associated with the nearest cluster in a $2S \times 2S$ neighborhood using the euclidean distance. For a given number of iterations, cluster centers are updated and each pixel is reassigned to a new cluster. Usually, few iterations are needed for the cluster centers to converge. For more details, we refer the reader to [Achanta et al., 2012]. Compared to other superpixels generation methods, SLIC is at the same time simple, memory efficient and fast. In contrast to the preprocessing technique based on k-means clustering, the SLIC segmentation is computed once and for all.

4.2 Numerical experiments

The objectives of the experiments are to:

- Empirically demonstrate that the preprocessing methods dramatically reduce the active learning calculation time without significant loss of effectiveness,

- To compare the different preprocessing techniques and the influence of the number of superpixels with regard to their impact on the AL performances.

4.2.1 Data

We conduct experiments with the Core-set technique on Indian Pines because Core-set is the technique that performed best in our experiments but that has large computational requirements. We choose the Indian Pines data set because it is small enough to run Core-set without preprocessing. In order to study the impact of preprocessing on an image with a higher spatial resolution, we also conducted experiments with Breaking Ties, which has low computational requirements, on Pavia University.

4.2.2 Results

Impact of the preprocessing on the computation time If the decrease of the query time for Breaking Ties is not interesting with regard to the additional segmentation time, major time gains are achieved for Core-set as table 4.5 shows.

Table 4.5: Approximate time requirements in minutes with our hardware[1].

Number of regions	**∅**	**20000**	**10000**	**5000**
		k-means		
Segmentation time	-	~ 25	~ 5	~ 1
		SLIC		
	-	~ 2.5	~ 1	< 1
		Core-set		
Query time	-	37	5	1
		Breaking Ties		
	0.050	0.017	0.012	0.010

Impact of the preprocessing on OA and mIoU Experiments on Pavia University with the Breaking Ties (see Fig. 4.15) technique show that 1) the preprocessing techniques have low impacts on the OA (at most 2% gap with AL without preprocessing) and 2) the choice of the preprocessing techniques has on average low influence on the OA. The same observations can be drawn concerning the mIoU score. Experiments on Indian Pines with the Core-set technique (see Fig. 4.15) do not show significant differences in terms of OA and mIoU between the runs with and without preprocessing either, nor between the preprocessing methods.

Impact of the number / size of superpixels Experiments on Pavia University with 1000 and 7500 superpixels show that slightly larger differences between preprocessing methods are observed with larger superpixels. In particular, subsampling reaches slightly lower OA and mIoU at first steps when only 1000 samples are drawn. Note that when 7500 superpixels are used, respectively 1000, the average size of a superpixel is 4 pixels, respectively 30 pixels.

4.3 Discussion

Experiments have shown that the superpixel-based preprocessing methods as well as subsampling allow to use computationally intensive active learning algorithms on large hyperspectral data sets without significant loss of effectiveness. This allows the use of AL methods that would normally have long calculation time during field campaigns, facilitating the annotation of experts. An expected advantage of k-means over SLIC is that it can gather pixels from arbitrarily large and non-adjacent regions that are spectrally homogeneous in one superpixel only, as we can observe in Fig. 4.14, which should avoid redundant queries, though the benefits are not clear in the experiments. Meanwhile, segmenting the pool with k-means at each step is more time consuming.

Unsurprisingly, experiments indicate that the loss of information using subsampling is higher than the loss of information with the superpixel segmentation when the subsampling rate is low (*i.e.* a small subset of the unlabeled pool is kept). We guess that even with large superpixels, the spectral average of the superpixels preserves useful information for the query system.

We believe that superpixel-based preprocessing may increase the robustness to outliers. As a matter of fact, considering individual pixels is prone to outliers selection because they can exhibit anomalous spectral signatures that yield high uncertainty. On the contrary, outliers spectral signatures are mixed with in-distribution samples when we use superpixels. Of course, the preprocessing will be detrimental to AL if isolated pixels are of interest for the user.

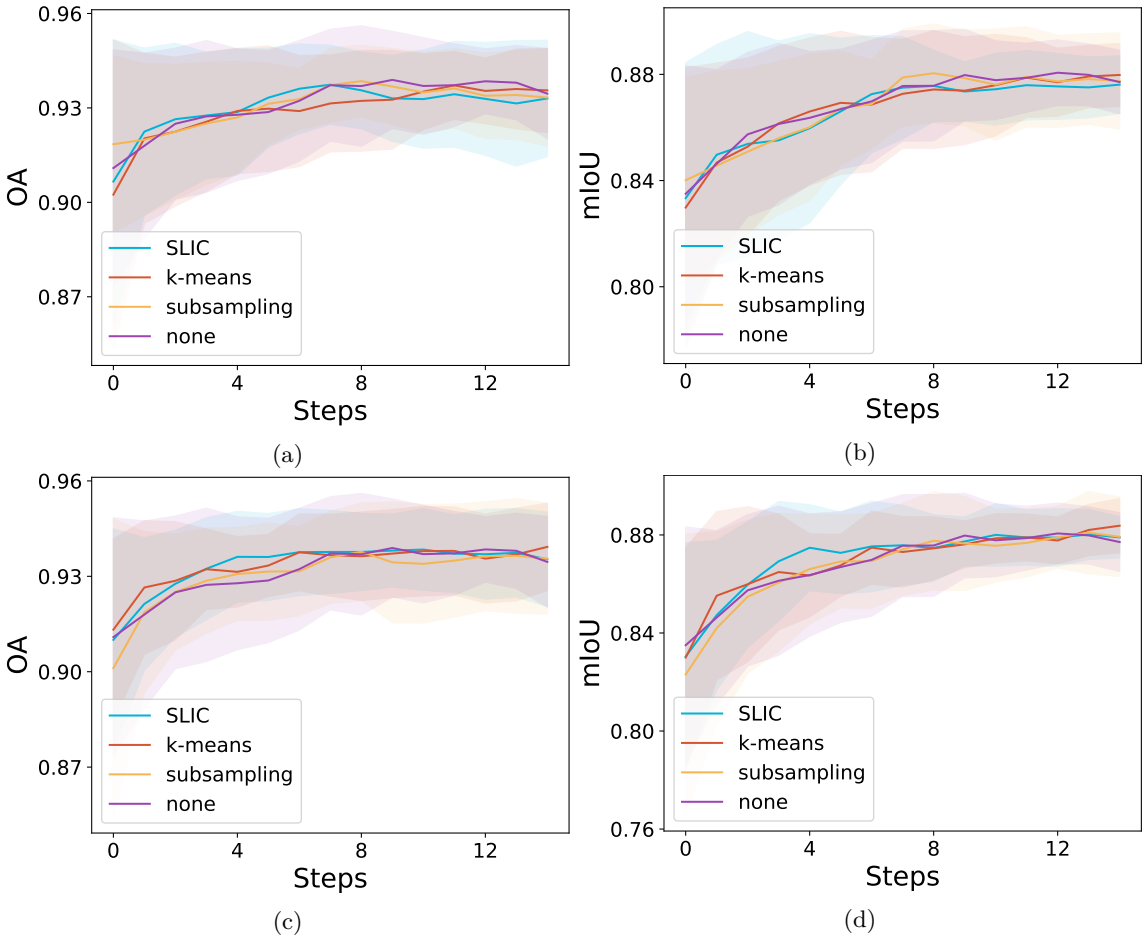


Figure 4.15: Accuracy metrics over a Breaking Tie AL process with different preprocessing techniques on Pavia University. The unlabeled pool is reduced to (a-b) 7500 superpixels and (c-d) 1000 superpixels.

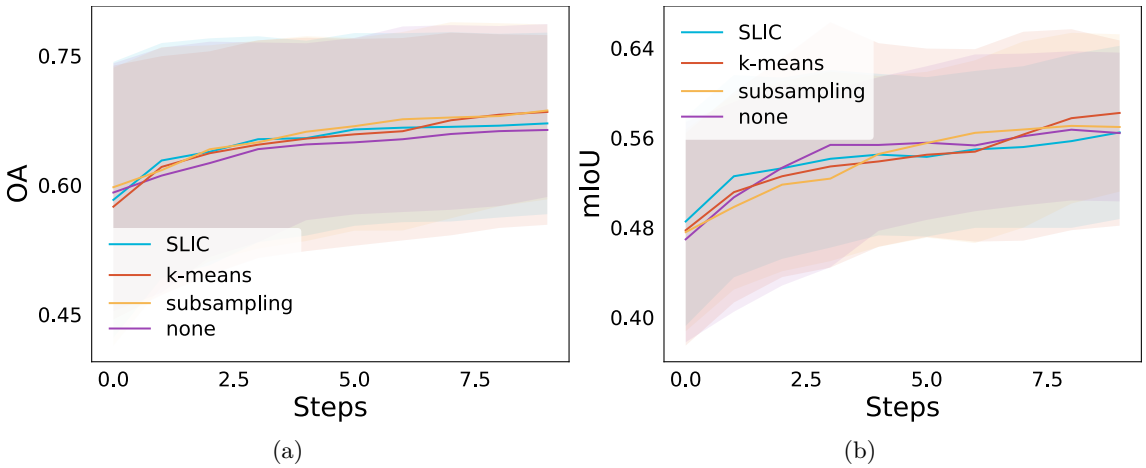


Figure 4.16: Accuracy metrics over a Core-set AL process with different preprocessing techniques on Indian Pines. The unlabeled pool is reduced to 1850 superpixels.

Choosing the right number of superpixels is not straightforward. There is a trade-off between large superpixels that decrease the computation time and redundancy at the expense of a higher loss of information and randomness, and small superpixels that have a lower impact on the calculation requirements. In addition, the appropriate size of superpixels in average depends on the nature of the landscapes. Notably, we believe that further experiments in dense urban areas could provide insights on the appropriate size of superpixels.

Finally, using more complex superpixel features might be an interesting improvement (additional spectral features or spatial features...) and selecting pixels at the center of the superpixel only would decrease the risk of drawing mixed pixels. Instead of taking random pixels within the superpixels, we could also run new active learning steps on the pixels of every selected superpixels and eventually combine different AL methods (one AL would be applied on the superpixels and another would be applied on the pixels).

5 Integrating a priori semantic knowledge

One limitation of AL methods is that they give equal importance to each class. However, in the context of impermeable surfaces classification, we have seen that some classes are more important than others. For instance, we prefer to well recognize impermeable roads from permeable railways rather than healthy grass from stressed grass. Therefore, we would like to avoid the waste of annotations on pixels that provide little information about what matters the most. In this section, we introduce Probabilistic Breaking Ties (PBT), an AL technique that builds on the state-of-the-art Breaking Ties (BT) method, to leverage the a priori semantic knowledge of the nomenclature.

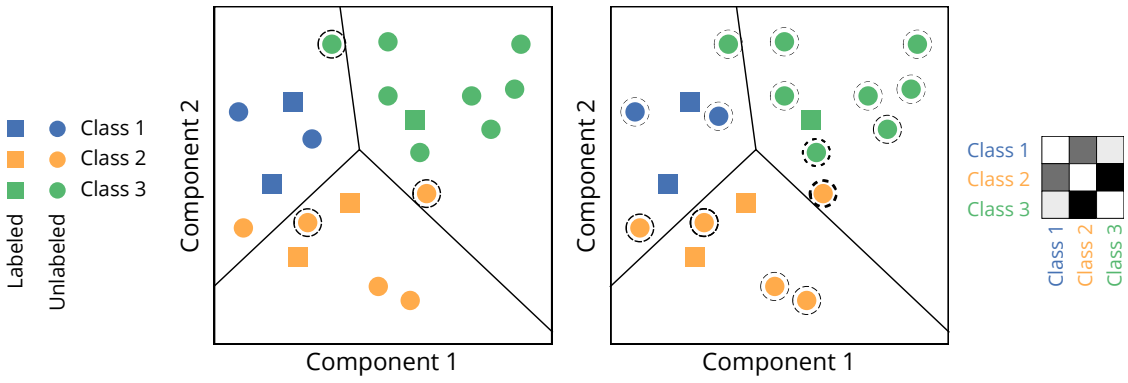
In the context of classification, leveraging class hierarchy has recently once again become an active research area. Recent works suggest to use cascades of neural networks to first predict coarse classes and then more subtle subclasses [Roy et al., 2020; Yan et al., 2015]. [Bertinetto et al., 2020] introduced two techniques to integrate the class hierarchy into training: a hierarchical cross-entropy with one-hot labels and a conventional cross-entropy with soft labels. Soft labels take into account the semantic similarities by using the distances between classes in the hierarchy. Formally, the soft label y_A^{soft} of the class A is defined for the coordinate corresponding to the class C as follows:

$$y_A^{soft}(C) = \frac{\exp(-\beta d(A,C))}{\sum_{B\in\mathcal{C}} \exp(-\beta d(B,C))} \tag{4.20}$$

where $\mathcal{C}$ is the set of classes, $\beta \geq 0$ is a hyperparameter and the distance $d(A,C)$ is defined as the length of the shortest path between A and C in the hierarchy. This semantic similarity measure is a common metric within ontology-based measures of semantic similarity [Bin et al., 2009]. Other common ontology-based metrics can be used when two nodes (*e.g.* concepts, classes...) are linked by several edges [Bin et al., 2009], or in the case where subclasses may be at different depths in the hierarchy [Lee et al., 2008].

5.1 Probabilistic Breaking Ties: method

Probabilistic Breaking Ties is based on the hierarchy between classes in the nomenclature. A land cover nomenclature can indeed be formalized as a tree where the leaves are the semantic classes. Nodes in the tree are larger categories. Like in [Landrieu and Garnot, 2021] and [Bertinetto et al., 2020], we use the distance between leaves to encode the semantic similarity



(a) Breaking Ties (b) Probabilistic Breaking Ties

Figure 4.17: Illustration of Breaking Ties and Probabilistic Breaking Ties on a 2D toy data set. On the left figure, the three black dotted circles around colored dots represent the unlabeled points that would be selected by Breaking Ties. On the right figure, the width of the black dotted circles represent the probability that an unlabaled data point would be selected by Probabilistic Breaking Ties: the larger the line width, the higher the probability. At the far right, the 3×3 matrix represent the cost of confusion between the classes: the darker the matrix at i^{th} row and j^{th} coordinate, the higher the cost of confusion between classes i and j.

between classes. The lower the distance, the higher the similarity. Similarly, the higher the distance, the higher the cost of confusion, which allows us to define a cost matrix, for C classes, as follows:

$$\forall (k,l) \in \{1,\dots,C\}^2, k \neq l,$$
$$D[k,l] = 10^{-\frac{\max_{u,v} d(u,v) - d(k,l)}{\beta}} \tag{4.21}$$

where $\beta > 0$ and $d(k,l)$ is the distance in the tree between the leaves k and l. More precisely, $d(k,l)$ is the number of nodes within the path from leave k to the closest common parent with leave l. For instance, referring to the Houston nomenclature illustrated in Fig. 4.18, we have that $d(\textit{Healthy grass}, \textit{Bare earth}) = 2$ and $d(\textit{Healthy grass}, \textit{Roads}) = 3$. We can note that $d(k,l) = d(l,k)$ because every leaves are at the same depth in the tree. The higher the coefficient β, the less important is the distance between leaves. This cost matrix is the core element of PBT. The idea is, instead of selecting b pixels with the highest Breaking Ties scores, to sample b points according to a distribution defined by a weighted version of the Breaking Ties scores, where the weights are defined by the cost matrix. The principle of PBT against BT is illustrated in Fig. 4.17.

Let's denote $X = \{x_i\}_{i \in (1,\dots,N)}$ a pool of N points. We define the PBT acquisition function over a set of points as follows:

$$a_{PBT}(X) = \{\frac{w_i \cdot p_i}{\sum_{k=1}^{N} w_k \cdot p_k}\}_{i \in (1,\dots,N)} \tag{4.22}$$

where, $\forall i \in (1,\dots,N)$,

$$p_i = 1 + a_{breaking-ties}(x_i) \tag{4.23}$$
$$w_i = \delta_i D[k,l] \tag{4.24}$$
$$k,\ l = \underset{i \in (1,\dots,c)}{\arg\max}\, f(x)^i,\ \underset{j, j \neq i}{\arg\max}\, f(x)^j \tag{4.25}$$
$$\delta_i = \begin{cases} 1 \text{ if } p_i > \gamma \\ 0 \text{ else} \end{cases} \tag{4.26}$$

with γ a threshold that controls the informativeness of the samples and D is the cost matrix.

Equation 4.23 defines p_i as a normalized version of the Breaking Ties score between 0 and 1: a high BT score leads to a high probability to select the pixel. Equation 4.24 defines w_i so that it is lower when the two most likely classes have a low confusion cost. Equation 4.26 defines δ_i so that a pixel with a very high Breaking Ties score (*i.e.* not informative), cannot be selected despite the re-weighting. In a nutshell, if $\beta = 1$, a sample $\boldsymbol{x}_1$ with the same Breaking Ties score than a sample $\boldsymbol{x}_2$ would be 10 times less likely to be sampled if the distance between the two most likely classes of $\boldsymbol{x}_1$ were closer by a distance of 1 in the tree than those of $\boldsymbol{x}_2$. If $\beta \longrightarrow +\infty$, then unlabeled data would be sampled taking into account only the Breaking Ties score.

5.2 Numerical experiments

In section 3.3, we argued that BT is particularly relevant to use after using representativeness AL techniques. The aim of PBT is not to compete with representativeness heuristics, but to enhance BT in its aim to refine class boundaries between similar classes. Therefore, we only compared the performances of PBT against BT in the following experiments.

5.2.1 Experimental settings

■ **Data**

Experiments were made on the Houston data set because it is, to our knowledge, the only public hyperspectral data set with as many classes that can be hierarchically organized. We arranged the nomenclature hierarchically, as Fig. 4.18 shows. We only kept in the initial training data set 200 labeled pixels per class, *i.e.* 4000 pixels in total, which is representative of an operational use case. Ten sets of different initial training data sets, unlabeled pools and test data sets were selected from disjoint regions.

■ **Validation** To compute the BT and the PBT scores, we used a classic spectral CNN [Hu et al., 2015] that was benchmarked in the comparative review [Audebert et al., 2019]. A better model would give better results, but we argue that the relative difference would be the same for BT and PBT. We performed 100 epochs with a 0.001 learning rate and a cross-entropy loss optimized through a stochastic gradient descent with momentum. At each step of the AL process (15 steps in total), we computed the following metrics based on the predictions of the CNN: the overall accuracy, the mean intersect over union and the average cost. The average cost (AC) is the weighted average of the number of confusions by their cost:

$$AC = \frac{1}{C^2} \frac{\sum_{i,j} Q[i,j] \cdot D[i,j]}{\sum_{i,j} D[i,j]} \tag{4.27}$$

where Q is the confusion matrix. We normalize the average cost by the sum of the confusion cost matrix so that it is invariant to the costs scale. OA and mIoU were computed for N to N classification but also for permeable versus impermeable classification (P/IP). In both cases, the model made predictions over C classes. Only then, the predicted labels were converted to *Permeable*, *Impermeable* or *Other*. In our experiments, we set $\beta = 1$. This means that for two pixels whose Breaking Ties scores are equal, the pixel at the frontier of two classes closer from one node are ten times less likely to be queried. We set $\gamma = 0.8$ so that only uncertain pixels are queried.

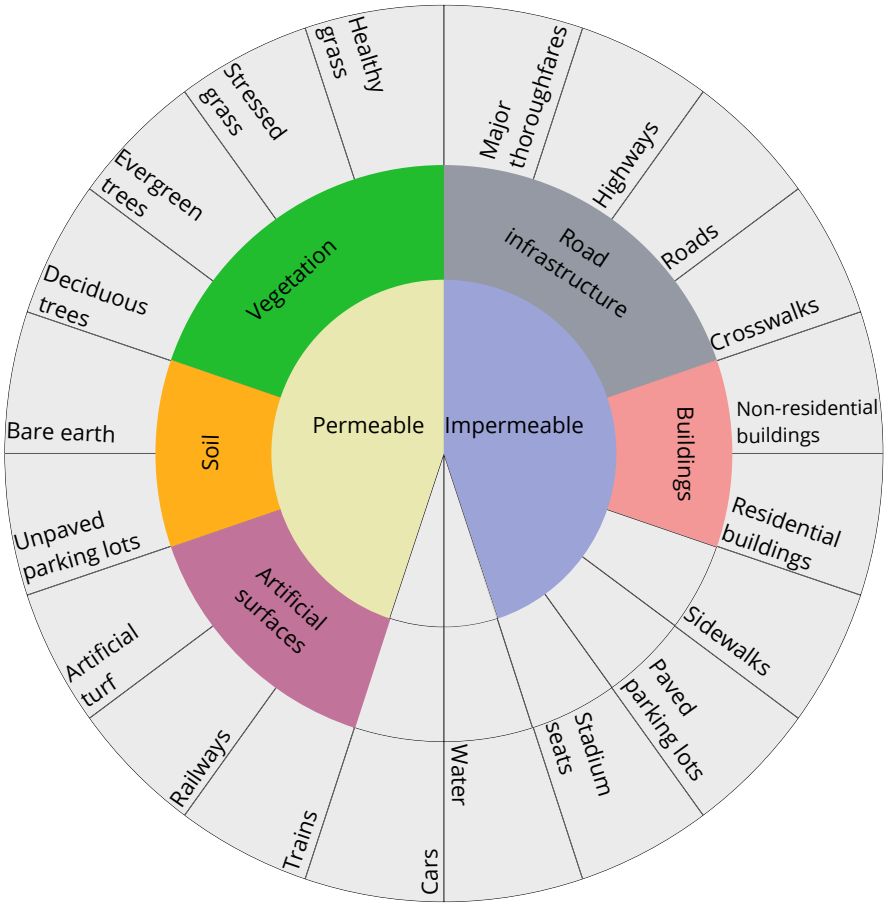
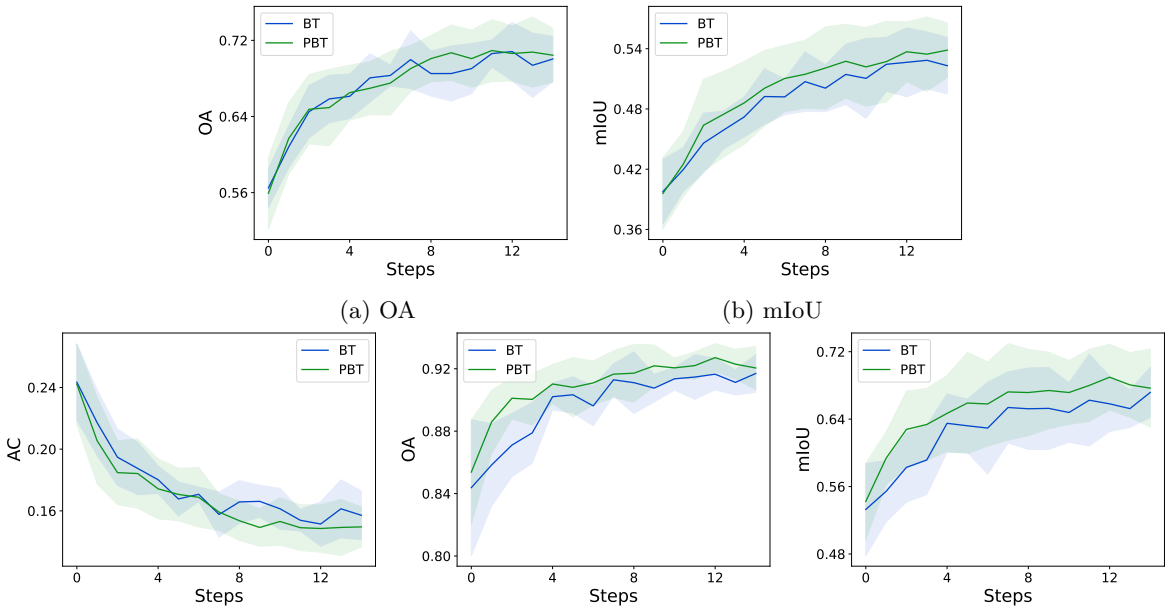


Figure 4.18: Our hierarchical organization of the nomenclature of the Houston data set. The center of the pie corresponds to the root of the tree representation, while the classes in the outer grey disk represent the leaves.



(a) OA (b) mIoU

(c) AC (d) P/IP OA (e) P/IP mIoU

Figure 4.19: Mean and standard deviation of (a) N vs N OA, (b) N vs N mIoU, (c) AC, (d) P/IP OA and (e) P/IP mIoU for permeable VS impermeable classification.

5.2.2 Results

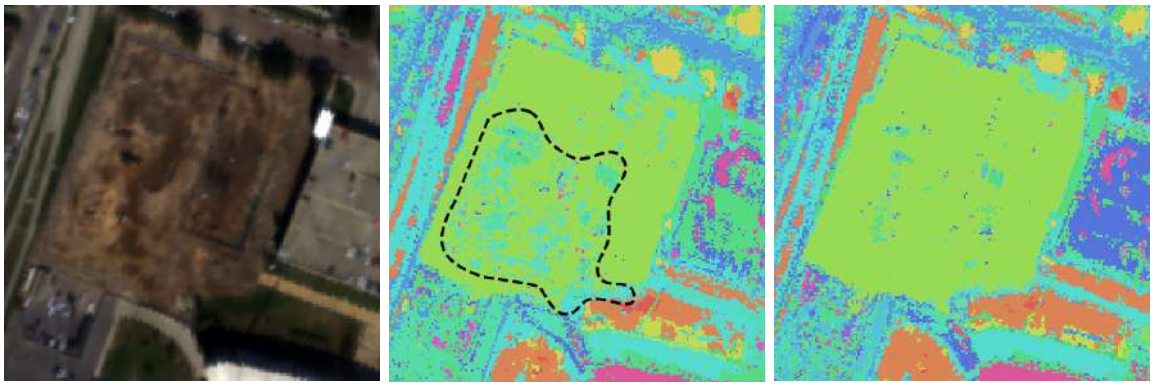
Fig. 4.19 shows that OA is barely unchanged between BT and PBT. On the contrary, mIoU was slightly increased by PBT, which was quite unexpected. It appears that PBT focused particularly on classes with high confusions rate.

Fig. 4.19 shows that PBT reached lower AC, especially at the beginning of the AL process. Similarly, impermeable vs permeable OA and mIoU were significantly increased by PBT. More specifically, PBT had a 4% higher P/IP mIoU at step 3. Moreover, PBT took two times less iterations than BT (and therefore two times less pixels to label) in order to reach a 0.64 P/IP mIoU score.

Fig. 4.20 shows land cover maps obtained at step 3 with the training data sets acquired with BT and PBT. With BT, many pixels of *bare earth* are confused with *sidewalks*, probably because their spectral signatures share similarities. The number of confusions are much fewer with PBT who focused between permeable and impermeable classes that are difficult to distinguish.

5.3 Discussion

Probabilistic Breaking Ties is an easy to implement improvement of Breaking Ties when the hierarchy of the nomenclature contains important semantic information. In an operational context, it allows to reduce the number of pixels to label, hence the duration of field campaigns, in order to reach good performances. One limitation of Probabilistic Breaking Ties however is its additional hyperparameters that may be difficult to properly tune. Besides, in future work, we will further study the impact of the stochasticity of Probabilistic Breaking Ties on the performance of the AL process. Finally, while we have integrated the class hierarchy into the AL acquisition function, another strategy could be to use the conventional



(a) (b) Breaking Ties (c) Probabilistic Breaking Ties

Figure 4.20: (a) False-color composition of a subset of the Houston image (b-c) land cover maps predicted with a CNN trained on the training set enriched by (b) 3 Breaking Ties queries and (c) 3 Probabilistic Breaking Ties queries. On the land cover (b), many pixels of *bare earth* in green are confused with *sidewalks* in light blue while the number of confusions are much less on the land cover map (c).

Breaking Ties technique and to post-process its results based on preference rules derived from the hierarchy.

6 Conclusions and perspectives

■ **Conclusions** In this chapter, we identified three techniques through numerical experiments, namely Core-set [Sener and Savarese, 2018], Breaking Ties [Tong Luo et al., 2004] and BALD [Houlsby et al., 2011], that can significantly improve the mapping of large hyperspectral images with few hundreds additional labeled pixels (by discovering important classes missing to the initial nomenclature and selecting pixels informative of the intra-class variability). If we had to choose only one, we would use Core-set because it can grasp the diversity of data in few AL steps while significantly increasing the segmentation performances (up to +13% accuracy with 300 additional labeled pixels compared to random sampling). We suggest to use Breaking Ties, or our improved version Probabilistic Breaking Ties for hierarchical data sets, only for the last AL iterations in order to raise ambiguities between similar classes. Besides, we argue that BALD should be specifically use to discover rare classes in the image. In that way, we believe that the combination of AL methods should be data-specific and that additional experiments in that direction would be interesting. The limitation of Core-set is that it requires data preprocessing techniques to be run in a reasonable time during field campaigns, though our experiments seem to indicate that preprocessing has no significant impacts on its performance.

■ **Perspectives** Superpixel-based preprocessing techniques could be interesting to deal with redundancy, outliers and mixed pixels, though the benefits are hard to demonstrate on benchmark data sets that contain barely any outliers or mixed pixels.

Moreover, we developed a QGIS[2] plug-in, AL4EO, in order to use AL techniques in an operational case: https://github.com/Romain3Ch216/AL4EO. We believe that, for AL4EO to be game-changing when doing field campaigns, more sophisticated or efficient AL techniques are not necessary, but that some simple features should be developed to assist the scientists or engineers on field:

[2] https://www.qgis.org/

- a Breaking Ties implementation for 1 VS all setting,
- instantaneous visualization tools of hyperspectral data,
- selection tools to quickly perform an AL step on a subset of an image,
- the joint processing of multiple images,
- a tool to select the closest spectra given a reference spectrum and / or hand-crafted spectral features.

Finally, the capacity of AL algorithms is limited, by nature, to the capacity of their associated machine learning models to learn robust representations. Ideally, AL query systems should only select pixels that are informative of the spectral semantic intra-class variability, as much as the intrinsic and physics variabilities should be more easy to capture by a machine learning model, provided appropriate architectures and learning algorithm. This is the reason why we study in the next chapter the possibility to improve the representation learning capacities of machine learning models by leveraging a priori physical knowledge.

The work presented in this chapter has resulted in a publication in one peer-review journal and two proceedings in international conferences:

· R. Thoreau, V. Achard, L. Risser, B. Berthelot and X. Briottet, "Active Learning for Hyperspectral Image Classification: A comparative review," in IEEE Geoscience and Remote Sensing Magazine, vol. 10, no. 3, pp. 256-278, Sept. 2022, doi: 10.1109/MGRS.2022.3169947,

· R. Thoreau, V. Achard, L. Risser, B. Berthelot and X. Briottet, "Active Learning On Large Hyperspectral Datasets: A preprocessing method", Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci., XLIII-B3-2022, 435–442, doi: 10.5194/isprs-archives-XLIII-B3-2022-435-2022,2022,

· R. Thoreau, V. Achard, L. Risser, B. Berthelot and X. Briottet, "Probabilistic Breaking Tie: An Active Learning Strategy To Leverage Class Hierarchy For Impervious Surfaces Classification," 2022 12th Workshop on Hyperspectral Imaging and Signal Processing: Evolution in Remote Sensing (WHISPERS), Rome, Italy, 2022, pp. 1-5, doi: 10.1109/WHISPERS56178.2022.9955057.

7 References

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., and Süsstrunk, S. (2012). Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2274–2282. 97, 99

Atighehchian, P., Branchaud-Charron, F., Freyberg, J., Pardinas, R., and Schell, L. (2019). Baal, a bayesian active learning library. https://github.com/ElementAI/baal/. 87

Audebert, N., Le Saux, B., and Lefèvre, S. (2018). Generative adversarial networks for realistic synthesis of hyperspectral samples. In *IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium*, pages 4359–4362. IEEE. 73

Audebert, N., Le Saux, B., and Lefèvre, S. (2019). Deep learning for classification of hyperspectral data: A comparative review. *IEEE Geoscience and Remote Sensing Magazine*, 7(2):159–173. 85, 86, 104

Baram, Y., El-Yaniv, R., and Luz, K. (2003). Online choice of active learning algorithms. In *Journal of Machine Learning Research - JMLR*, volume 5, pages 19–26. 76

Bertinetto, L., Mueller, R., Tertikas, K., Samangooei, S., and Lord, N. A. (2020). Making better mistakes: Leveraging class hierarchies with deep networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12506–12515. 102

Bin, S., Liying, F., Jianzhuo, Y., Pu, W., and Zhongcheng, Z. (2009). Ontology-based measure of semantic similarity between concepts. In *2009 WRI World Congress on Software Engineering*, volume 2, pages 109–112. IEEE. 102

Cai, L., Xu, X., Liew, J. H., and Foo, C. S. (2021). Revisiting superpixels for active learning in semantic segmentation with realistic annotation costs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10988–10997. 97

Cook, W. J., Cunningham, W. H., Pulleyblank, W. R., and Schrijver, A. (1998). *Combinatorial optimization*, volume 605. Springer. 80

Dasgupta, S. and Hsu, D. (2008). Hierarchical sampling for active learning. In *Proceedings of the 25th international conference on Machine learning*, pages 208–215. 72, 76, 77, 80

Demir, B., Persello, C., and Bruzzone, L. (2011). Batch-mode active-learning methods for the interactive classification of remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 49(3):1014–1031. 75

Di, W. and Crawford, M. M. (2010). Multi-view adaptive disagreement based active learning for hyperspectral image classification. In *2010 IEEE International Geoscience and Remote Sensing Symposium*, pages 1374–1377. 75

Ducoffe, M. and Precioso, F. (2018). Adversarial active learning for deep networks: a margin based approach. *arXiv preprint arXiv:1802.09841*. 75

Ferecatu, M. and Boujemaa, N. (2007). Interactive remote-sensing image retrieval using active relevance feedback. *Geoscience and Remote Sensing, IEEE Transactions on*, 45:818 – 826. 74

Freeman, L. G. (1965). *Elementary applied statistics*. 74

Gal, Y. (2016). *Uncertainty in Deep Learning*. PhD thesis, University of Cambridge. 75, 77

Gal, Y. and Ghahramani, Z. (2016). Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR. 75, 85

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*. 75

Houlsby, N., Huszár, F., Ghahramani, Z., and Lengyel, M. (2011). Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*. 72, 75, 77, 78, 107

Hsu, W.-N. and Lin, H.-T. (2015). Active learning by learning. In *AAAI*. 76

Hu, W., Huang, Y., Wei, L., Zhang, F., and Li, H. (2015). Deep convolutional neural networks for hyperspectral image classification. *Journal of Sensors*, 2015:1–12. 85, 104

Kasarla, T., Nagendar, G., Hegde, G. M., Balasubramanian, V., and Jawahar, C. (2019). Region-based active learning for efficient labeling in semantic segmentation. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1109–1117. 97

Kellenberger, B., Marcos, D., Lobry, S., and Tuia, D. (2019). Half a percent of labels is enough: Efficient animal detection in uav imagery using deep cnns and active learning. *IEEE Transactions on Geoscience and Remote Sensing*, 57(12):9524–9533. 73

Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization. In Bengio, Y. and LeCun, Y., editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. 85

Kingma, D. P. and Welling, M. (2014). Auto-encoding variational bayes. In Bengio, Y. and LeCun, Y., editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*. 75

Kirsch, A., Van Amersfoort, J., and Gal, Y. (2019). Batchbald: Efficient and diverse batch acquisition for deep bayesian active learning. *Advances in neural information processing systems*, 32. 72, 75, 77, 78, 79

Konyushkova, K., Sznitman, R., and Fua, P. (2017). Learning active learning from data. *Advances in neural information processing systems*, 30. 72, 76, 77, 83

Landrieu, L. and Garnot, V. S. F. (2021). Leveraging class hierarchies with metric-guided prototype learning. In *British Machine Vision Conference (BMVC)*. 102

Lee, W.-N., Shah, N., Sundlass, K., and Musen, M. (2008). Comparison of ontology-based semantic-similarity measures. In *AMIA annual symposium proceedings*, volume 2008, page 384. American Medical Informatics Association. 102

Li, X. and Guo, Y. (2013). Adaptive active learning for image classification. In *Proceedings / CVPR, IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 859–866. 75

Lloyd, S. (1982). Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2):129–137. 97

Ranganath, R., Gerrish, S., and Blei, D. (2014). Black box variational inference. In *Artificial intelligence and statistics*, pages 814–822. PMLR. 75

Roy, D., Panda, P., and Roy, K. (2020). Tree-cnn: a hierarchical deep convolutional neural network for incremental learning. *Neural Networks*, 121:148–160. 102

Ruzicka, V., D'Aronco, S., Wegner, J. D., and Schindler, K. (2020). Deep active learning in remote sensing for data efficient change detection. *ArXiv*, abs/2008.11201. 73

Sener, O. and Savarese, S. (2018). Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*. 72, 75, 77, 79, 80, 81, 107

Settles, B. (2012). *Active Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan and Claypool Publishers. 72, 74, 75, 76

Seung, H., Opper, M., and Sompolinsky, H. (1992). Query by committee. In *COLT 92*. 75

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423. 74

Sinha, S., Ebrahimi, S., and Darrell, T. (2019). Variational adversarial active learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5972–5981. 72, 75, 77

Sobel, I. (2014). An Isotropic 3x3 Image Gradient Operator. *Presentation at Stanford A.I. Project 1968*. 87

Tipping, M. E. and Bishop, C. M. (1999). Mixtures of probabilistic principal component analyzers. *Neural computation*, 11(2):443–482. 85

Tong Luo, Kramer, K., Samson, S., Remsen, A., Goldgof, D. B., Hall, L. O., and Hopkins, T. (2004). Active learning to recognize multiple types of plankton. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, volume 3, pages 478–481 Vol.3. 72, 74, 77, 107

Tuia, D., Volpi, M., Copa, L., Kanevski, M., and Munoz-Mari, J. (2011). A survey of active learning algorithms for supervised remote sensing image classification. *IEEE Journal of Selected Topics in Signal Processing*, 5(3):606–617. 74, 75

Vincent, L. (1993). Grayscale area openings and closings, their efficient implementation and applications. In *First Workshop on Mathematical Morphology and its Applications to Signal Processing*, pages 22–27. 87

Wang, Z. and Ye, J. (2015). Querying discriminative and representative samples for batch mode active learning. *ACM Trans. Knowl. Discov. Data*, 9(3). 75

Wei, K., Iyer, R., and Bilmes, J. (2015). Submodularity in data subset selection and active learning. In Bach, F. and Blei, D., editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1954–1963, Lille, France. PMLR. 75

Wingate, D. and Weber, T. (2013). Automated variational inference in probabilistic programming. *arXiv preprint arXiv:1301.1299*. 75

Xie, B., Yuan, L., Li, S., Liu, C. H., and Cheng, X. (2022). Towards fewer annotations: Active learning via region impurity and prediction uncertainty for domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8068–8078. 74

Xie, S., Feng, Z., Chen, Y., Sun, S., Ma, C., and Song, M. (2020). Deal: Difficulty-aware active learning for semantic segmentation. In *Proceedings of the Asian conference on computer vision*. 74

Yan, Z., Zhang, H., Piramuthu, R., Jagadeesh, V., DeCoste, D., Di, W., and Yu, Y. (2015). Hd-cnn: hierarchical deep convolutional neural networks for large scale visual recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 2740–2748. 102

Yang, Y., Ma, Z., Nie, F., Chang, X., and Hauptmann, A. (2015). Multi-class active learning by uncertainty sampling with diversity maximization. *International Journal of Computer Vision*, 113. 75

Zhang, Z., Pasolli, E., and Crawford, M. M. (2020). An adaptive multiview active learning approach for spectral–spatial classification of hyperspectral images. *IEEE Transactions on Geoscience and Remote Sensing*, 58(4):2557–2570. 73

Zhao, W., Chen, X., Chen, J., and Qu, Y. (2020). Sample generation with self-attention generative adversarial adaptation network (sagaan) for hyperspectral image classification. *Remote Sensing*, 12(5):843. 73

Zhu, J.-J. and Bento, J. (2017). Generative adversarial active learning. *arXiv preprint arXiv:1702.07956*. 73

Chapter 5

Hybrid modeling: improving the representation of physics intra-class variability from physical prior knowledge

Contents

1 Chapter summary

Due to the various sources of spectral intra-class variability, machine learning models need many labeled pixels to learn representations that are robust to spectral variations. When data annotations are scarce, Active Learning techniques can leverage statistical heuristics to guide the choice of few additional pixels to label, which may significantly improve the *quality* of the training set as we have seen in chapter 4. Yet, *physics* intra-class variability, which is a consequence of the variations of the local irradiance conditions, can be analytically modeled. Hence, this a priori physical knowledge could circumvent the need of labeled samples to capture the physics-based variability. Hybrid modeling, that is the combination of physics with machine learning models, has recently raised a lot of attention, demonstrating promising properties such as improved interpolation and extrapolation capabilities and increased interpretability [Raissi et al., 2019; Takeishi and Kalousis, 2021]. Conventional machine learning models learn correlations, from a training data set, in order to map observations to targets or latent representations, with the hope to generalize to new data. In other words, machine learning models implicitly make assumptions on the real data distribution that are consistent with the training data distribution, *i.e.* inductive biases [Mitchell, 1980; Zhao et al., 2018], that usually do not generalize in small data regime [von Rueden et al., 2021]. In contrast, hybrid models are partially grounded on deductive biases, *i.e.* assumptions derived, in our context, from physics models that generalize, by nature, to out-of-distribution data.

Problematic

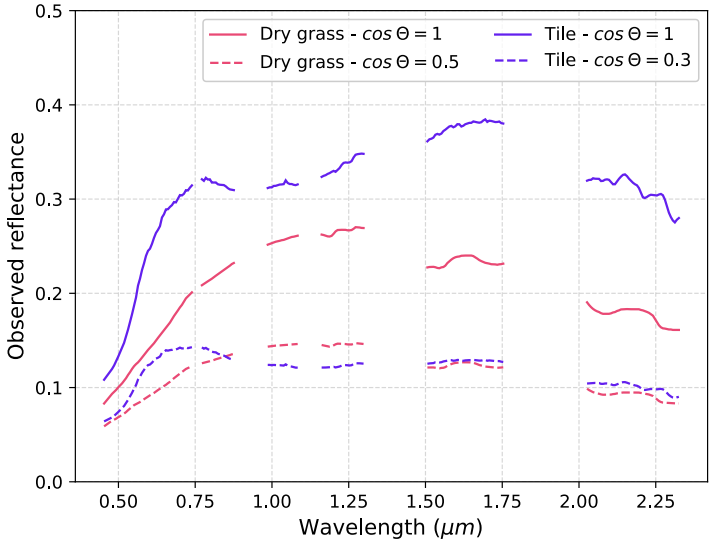
To what extent can a priori physical knowledge improve the robustness of spectral representations, learned by machine learning models, to *physics* intra-class variations in a context of a limited number of pixel annotations?

■ Summary of contributions

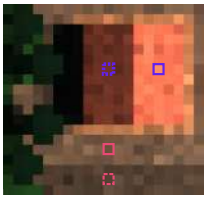
⋆ **Integration of a physical model in a VAE: a general framework** In section 3, we introduce p^{3}VAE, a variational autoencoder that integrates a physical model as part of its decoder. p^{3}VAE aims to decouple the variation factors that are intrinsic to materials from physical factors related to acquisition conditions. Integrating physics in an autoencoder was first proposed by [Aragon-Calvo and Carvajal, 2020]. They used a fully physical model inplace the decoder to inverse a 2D exponential light galaxy profile model. [Takeishi and Kalousis, 2021] generalized the work of [Aragon-Calvo and Carvajal, 2020] by developing a mathematical formalism introduced as physics-integrated VAEs. Physics-integrated VAEs (ϕ-VAE) complement an imperfect physical model with a machine learning model in the decoder of a VAE. To have a consistent use of physics despite the high representation power of machine learning models, they employ a regularization strategy that is central to their contribution. In contrast, p^{3}VAE integrates an assumed **p**erfect **p**hysics model that **p**artially captures the factors of variations of the data. In other words, we consider cases where the forward model is partially known : the physical model cannot approximate the data distribution by itself, but can accurately relate a subset of the underlying causes to the observations.

⋆ **Application of the hybrid model to the segmentation of hyperspectral images** As far as the irradiance conditions induce complex and non-linear data variations, we argue that a conventional generative model would hardly decorrelate intrinsic factors from physical factors (see Fig. 5.1 for an illustration of the physics-based variability). In section 2, we derive a simplified radiative transfer model that explains the *physics* intra-class variability as a function of the true irradiance conditions (that locally depend on topography) and the assumed irradiance conditions. In section 3.2, we apply the framework of p^{3}VAE to the

segmentation of hyperspectral data. The decoder of p^{3}VAE is a combination of the physical model with a neural network that map latent variables to hyperspectral data. In fact, the physics part of the decoder can be seen as a conventional dense layer whose weights and biases are derived from the physical model. The latent space of p^{3}VAE is implicitly divided into two subsets: one subset that is related to physical quantities and another one that has no physical meaning. This disentanglement is fostered by the physical model that takes as input a subset of the latent space as if it was a faithful physical quantity.



(a) Observed reflectance spectra



(b) False-color composition of the hyperspectral image

(c) 3D view of the scene

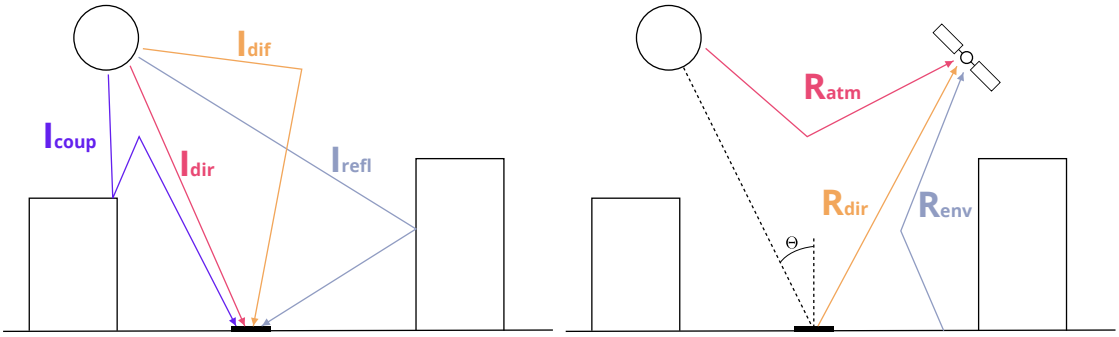
Figure 5.1: Illustration of intra-class variabilities and inter-class similarities due to different pixel-wise irradiance conditions. Θ is the solar zenith angle. The lower is Θ, the higher is the direct irradiance.

⋆ **Empirical study on simulated and real hyperspectral data** In order to evaluate p^{3}VAE in terms of accuracy, interpretability and disentanglement, we simulated hyperspectral data with exhaustive irradiance conditions. Moreover, to assess the validity of the simplified physics model in real conditions, and ultimately the potential of p^{3}VAE on real data, we annotated a real hyperspectral image. In particular, we compare the extrapolation and disentanglement performances of p^{3}VAE with a conventional semi-supervised VAE and a semi-supervised GAN-like model, as well as common semantic segmentation models.

2 Modeling physics intra-class variability

In order to understand the origins of the *physics* intra-class variability, we briefly recap the basics of radiative transfer modeling and the principles of the algorithms used to convert the raw data measured by the airborne or spaceborne sensors to ground reflectance images.

Remote sensing optical sensors measure spectral radiance, *i.e.* a radiant flux per unit solid angle, per unit projected area, per unit wavelength (W.sr^{-1}.m^{-2}.nm^{-1}). The acquired spectral radiance, in the reflective domain, is dependent of sun irradiance, of the atmospheric composition as well as the land cover and comes from various sources (see the illustration of irradiance and radiance terms on Fig. 5.2). In contrast, reflectance, that is the ratio of the reflected radiant flux on the incident radiant flux, only depends on matter chemical composition. Therefore, reflectance is a much more relevant feature for semantic segmentation. Atmospheric correction codes, such as COCHISE [Miesch et al., 2005], developed at ONERA, aim to deduct the atmospheric contribution to the measured radiance and to estimate pixel-



(a) Irradiance terms at surface level

(b) Radiance terms at sensor level

Figure 5.2: Illustration of the radiative components described in section 2. Figure reproduced from [Roussel et al., 2017].

wise reflectance. COCHISE, as many others atmospheric correction algorithms, assumes that land surfaces are lambertian, *i.e.* that they reflect radiation isotropically. We express the reflectance, as it is commonly done in the literature, at wavelength λ of a pixel of coordinates (x, y) as $\rho^{xy\lambda}$:

$$\rho^{xy\lambda} = \frac{\pi R_{dir}^{xy\lambda}}{I_{tot}^{xy\lambda} \tau_{dir}^{\lambda}} \tag{5.1}$$

with:

$$\begin{cases} R_{dir}^{xy\lambda} & = R_{tot}^{xy\lambda} - R_{env}^{xy\lambda} - R_{atm}^{\lambda} \\ I_{tot}^{xy\lambda} & = I_{dir}^{xy\lambda} + I_{dif}^{xy\lambda} + I_{coup}^{xy\lambda} + I_{refl}^{xy\lambda} \end{cases} \tag{5.2}$$

where (leaving the dependence to x, y and λ implicit):

- R_{tot} is the radiance measured by the sensor,
- R_{atm} is the portion of R_{tot} that is scattered by the atmosphere without any interaction with the ground,
- R_{env} is the portion of R_{tot} that comes from the neighbourhood of the pixel,
- R_{dir} is the portion of R_{tot} that comes from the pixel,
- I_{dir} is the irradiance directly coming from the sun,
- I_{dif} is the irradiance scattered by the atmosphere,
- I_{coup} is the irradiance coming from the coupling between the ground and the atmosphere,
- I_{refl} is the irradiance coming from neighbouring 3D structures.

Those terms are illustrated on Fig. 5.2. More precisely,

$$I_{dir} = \delta_{dir} \cdot cos\, \Theta \cdot I_{dir}^{o} \tag{5.3}$$

where δ_{dir} is the portion of pixel directly lit by sun and I_{dir}^{o} is the direct irradiance at surface level for $\delta_{dir} = 1$ and $\Theta = 0$. I_{dif} can be approximated by:

$$I_{dif} = \Omega \cdot p_{\Theta} \cdot I_{dif}^{o} \tag{5.4}$$

where Ω is the sky viewing angle factor, p_{Θ} is a correction coefficient that accounts for the anisotropy of the diffuse irradiance and I_{dif}^{o} is the diffuse irradiance for a pixel on a horizontal ground with $\Omega = 1$ (*i.e.* a full half sphere). As a matter of fact, the diffuse irradiance near the sun direction is much higher than the diffuse irradiance from directions further away

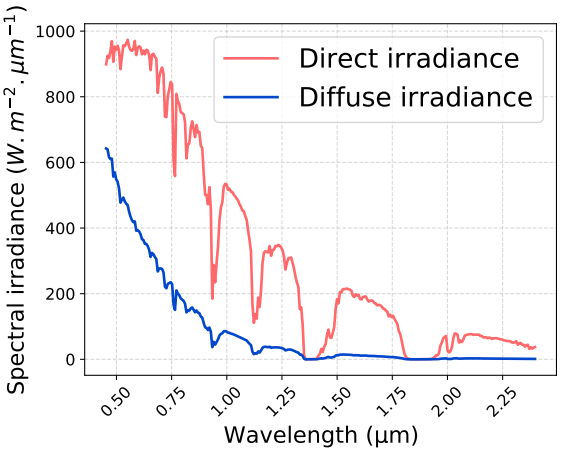


Figure 5.3: Illustration of spectral irradiances used by COCHISE to process the airborne images of Toulouse (see 3.3.2).

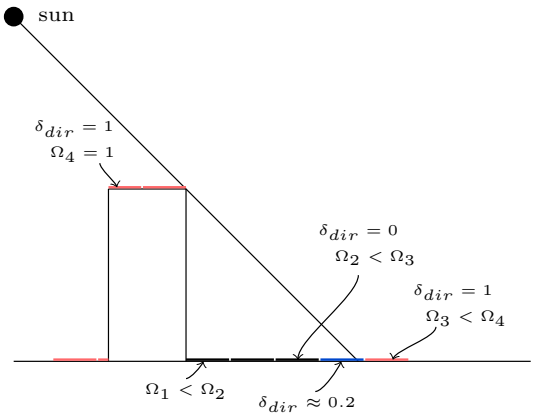


Figure 5.4: Illustration of the direct irradiance factor δ_{dir} and of the sky viewing angle factor Ω.

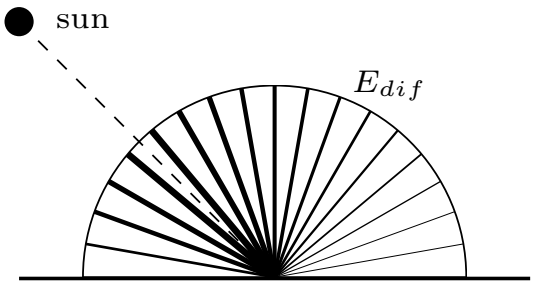


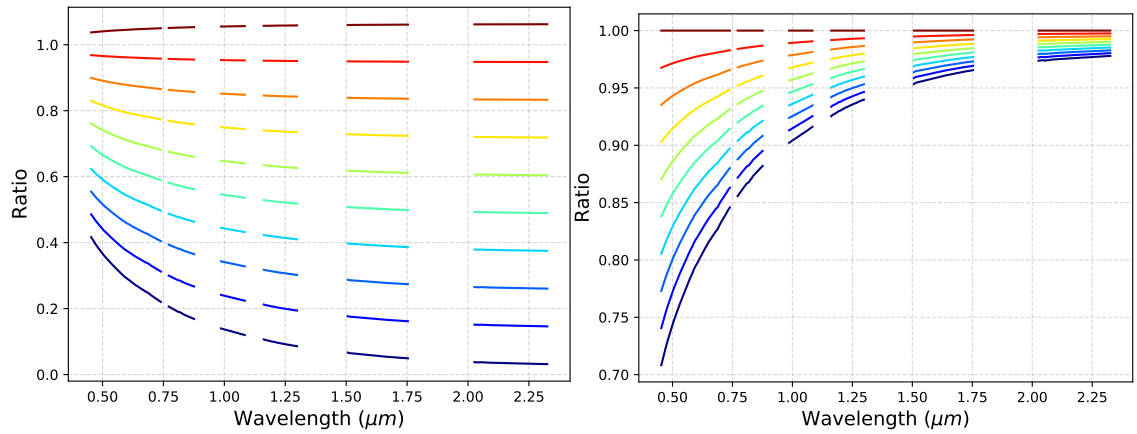
Figure 5.5: Illustration of anisotropy of the diffuse irradiance (the larger the rays the higher the diffuse irradiance, though it is nor proportional nor faithful to a realistic case).

from the sun, as illustrated in Fig. 5.5. Thus, the true diffuse irradiance depends on the part of the sky observed from the pixel point of view. We emphasize here that this is a very simplistic model that approximates an integral of products over the hemisphere as a product of integrals:

$$I_{dif} = \int_\theta \int_\phi \Omega(\theta,\phi) p(\theta,\psi) I^o_{dif} d\theta d\phi \approx I^o_{dif} \underbrace{\int_\theta \int_\phi \Omega(\theta,\phi) d\theta d\phi}_{\Omega} \underbrace{\int_\theta \int_\phi p(\theta,\psi) d\theta d\phi}_{p_\Theta} \int_\theta \int_\phi d\theta d\phi \tag{5.5}$$

where θ and ϕ denote the zenith and the azimuth angles in a spherical coordinate system, $\Omega(\theta,\phi)$ is equal to one if the sky in direction (θ,ϕ) is visible from the ground, zero otherwise, and $p(\theta,\phi)I^o_{dif}$ is the diffuse irradiance at coordinates (θ,ϕ). Different direct irradiance factors δ_{dir} and sky viewing angle factors Ω are illustrated in Fig. 5.4.

Atmospheric codes that do not need a digital surface model (topography and buildings data) such as COCHISE make the hypothesis that the ground is flat: $\Theta = \Theta^o$ (the solar zenith angle), $\delta_{dir} = 1$ (there are no shadows), and $\Omega = 1$ (each pixel sees the entire sky). Accordingly, pixels on a slope or in shadows will have poor reflectance estimates. More precisely, if we neglect the contributions of I_{coupl} and I_{refl}, we can easily derive the ratio between the



(a) $I_{dif} = I^o_{dif}$, $\delta_{dir} cos\,\Theta \in [0, 1]$ (b) $I_{dir} = I^o_{dir}$, $\Omega\, p_\Theta \in [0.3, 1]$

Figure 5.6: Ratio of the estimated reflectance under the true reflectance for varying irradiance conditions. Blue lines correspond to low values and red lines to high values of (a) $\delta_{dir} cos\,\Theta$ and (b) $\Omega\, p_\Theta$.

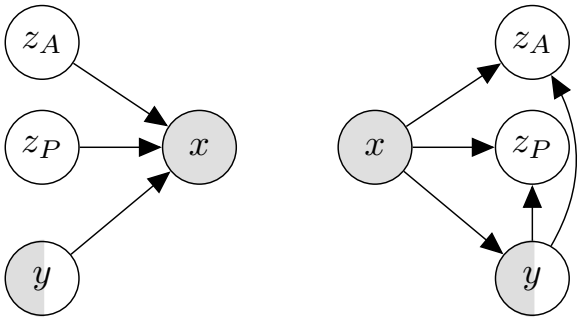
estimated reflectance ρ and the true reflectance ρ^* for a given wavelength, as a function of the true local parameters δ^*_{dir}, Θ^*, Ω^* and p^*_Θ:

$$\frac{\rho}{\rho^*} \approx \frac{\overbrace{I^*_{dir} + I^*_{dif}}^{\text{True irradiance}}}{\underbrace{I^o_{dir} + I^o_{dif}}_{\substack{\text{Assumed irradiance} \\ \text{by COCHISE}}}} = \frac{\delta^*_{dir} \cdot cos\,\Theta^* \cdot I^o_{dir} + \Omega^* \cdot p^*_\Theta \cdot I^o_{dif}}{cos\,\Theta^o \cdot I^o_{dir} + I^o_{dif}} \tag{5.6}$$

This ratio is a deductive bias that we would like to introduce in the training of a machine learning model. It is a strong assumption on how the signal behaves under local irradiance conditions. Here we emphasize the fact that the ratio depends on the wavelength λ and that it is non-linear with regard to δ^*_{dir}, $cos\,\Theta^*$, Ω^* and p^*_Θ, as illustrated on Fig. 5.6. The most non linear variations happen in the visible and the near infrared because the diffuse irradiance is the strongest in those spectral domains. On the contrary, because the diffuse irradiance is very low in the SWIR, variations of the direct irradiance cause translations of reflectance by a constant multiplicative factor in the SWIR while variations of diffuse irradiance have barely any impact on the reflectance in the SWIR. We argue that this behavior could hardly be modeled by a neural network that would likely fall in local optima.

3 Integrating physics into a VAE to improve spectral representation

In hyperspectral image semantic segmentation, we argue that a key problem is to distinguish the *intrinsic* and *semantic* variabilities from the *physics* variability, in a context of scarce annotations. Furthermore, we argue that probabilistic latent variable models are appropriate to learn meaningful data representations. In particular, we believe that good representations come along with a disentangled latent space, meaning that different factors of variations are encoded in different latent variables: *physics* , *intrinsic* and *semantic* intra-class variabilities should be encoded in three independent variables. Therefore, we introduced an hybrid model, called p^{3}VAE, to decouple the variation factors that are intrinsic to the data from



(a) Decoder (forward model)

(b) Encoder (inverse model)

Figure 5.7: Graphical model representations of (a) the likelihood $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}_A, \boldsymbol{z}_P, y)$ and (b) the variational posterior approximation $q_{\boldsymbol{\phi}}(y|\boldsymbol{x})q_{\boldsymbol{\phi}}(\boldsymbol{z}_A, \boldsymbol{z}_P|\boldsymbol{x}, y)$ of p^{3}VAE. White nodes are latent variables while gray ones are observed variables. Note that y can be either observed or not.

physical factors related to acquisition conditions. p^{3}VAE is a physics-integrated variational autoencoder that combines a physical model with a machine learning model. As much as disentanglement is a key property in many applications, we first present the general framework of p^{3}VAE in section 3.1. Then, we present how p^{3}VAE can be applied to hyperspectral image semantic segmentation in section 3.2. Third, we describe our numerical experiments on a simulated and a real hyperspectral data sets in section 3.3. Finally, we discuss the benefits and limitations of the model in section 3.4.

3.1 p^{3}VAE general framework

Our model builds on the concept of physics-integrated VAEs [Takeishi and Kalousis, 2021], that we will refer as ϕ-VAE. In this section, we describe the architecture of our model, its optimization scheme, inference procedure and discuss the related work in hybrid modeling.

3.1.1 p^{3}VAE architecture

We assume that a data point $\boldsymbol{x} \in \mathcal{X}$ is generated by a random process that involves continuous latent variables $\boldsymbol{z}_A \in \mathcal{Z}_A$ and $\boldsymbol{z}_P \in \mathcal{Z}_P$ as well as discrete variables $y \in \mathcal{Y}$ that are sometimes observed, and sometimes unobserved. We assume that the latent variable $\boldsymbol{z}_P$ is grounded to some physical properties while the latent variable $\boldsymbol{z}_A$ has no physical meaning. $\boldsymbol{z}_P$ could be, for instance, the pixel-wise irradiance mentioned in section 2. y is a categorical variable such as the class, for which we have annotations for a few data points. $\mathcal{X}$, $\mathcal{Y}$, $\mathcal{Z}_A$ and $\mathcal{Z}_P$ are subsets of the euclidean space.

The generative process consists of two steps: (1) values of $\boldsymbol{z}_A$, $\boldsymbol{z}_P$ and y are generated from a prior distribution $p(\boldsymbol{z}_A, \boldsymbol{z}_P, y)$; (2) a value $\boldsymbol{x}$ is generated from the conditional distribution $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}_A, \boldsymbol{z}_P, y)$ parameterized by $\boldsymbol{\theta}$. We assume that latent variables $\boldsymbol{z}_A$, $\boldsymbol{z}_P$, y are independent factors of variations, *i.e.* that they are mutually independent:

$$p(\boldsymbol{z}_A, \boldsymbol{z}_P, y) := p(\boldsymbol{z}_A)p(\boldsymbol{z}_P)p(y) \tag{5.7}$$

Besides, we assume a Gaussian observation noise Σ and define the likelihood as follows:

$$p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}_A, \boldsymbol{z}_P, y) := \mathcal{N}(\boldsymbol{x}|f_P \circ f_A(\boldsymbol{z}_A, \boldsymbol{z}_P, y), \Sigma) \tag{5.8}$$

where $f_A : \mathcal{Z}_A \times \mathcal{Y} \longrightarrow \mathcal{X}$ is a neural network and $f_P : \mathcal{Z}_P \times \mathcal{X} \longrightarrow \mathcal{X}$ is a physics model differentiable with regard to its inputs. We assume that the physics model f_P is perfect in the sense that it can deterministically represent the variation of the data induced by the variation of some physical quantity $\boldsymbol{z}_P$.

We would like to maximize the marginal likelihood $p_{\boldsymbol{\theta}}(\boldsymbol{x}, y)$ when y is observed and $p_{\boldsymbol{\theta}}(\boldsymbol{x})$ otherwise, which are unfortunately intractable. Therefore, we follow the state-of-the-art variational optimization technique introduced in [Kingma and Welling, 2014] by approximating the true posterior $p_{\boldsymbol{\theta}}(\boldsymbol{z}_A, \boldsymbol{z}_P, y|\boldsymbol{x})$ by a recognition model $q_{\boldsymbol{\phi}}(\boldsymbol{z}_A, \boldsymbol{z}_P, y|\boldsymbol{x})$. Furthermore, we assume that $q_{\boldsymbol{\phi}}(\boldsymbol{z}_A, \boldsymbol{z}_P|\boldsymbol{x}, y)$ factorizes as follows:

$$q_{\boldsymbol{\phi}}(\boldsymbol{z}_A, \boldsymbol{z}_P|\boldsymbol{x}, y) := q_{\boldsymbol{\phi}}(\boldsymbol{z}_A|\boldsymbol{x}, y) q_{\boldsymbol{\phi}}(\boldsymbol{z}_P|\boldsymbol{x}, y) \tag{5.9}$$

The encoding part $q_{\boldsymbol{\phi}}(\boldsymbol{z}_A, \boldsymbol{z}_P, y|\boldsymbol{x})$ and the decoding part $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}_A, \boldsymbol{z}_P, y)$ form a variational autoencoder (with parameters $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$).

3.1.2 p^{3}VAE semi-supervised training

We now explain how we adapted the semi-supervised optimization scheme introduced by [Kingma et al., 2014] to the training of our model.

Semi-supervised Model Objective. The objective function derived by [Kingma et al., 2014] for an observation x is twofold. First, when its label y is observed, the evidence lower bound of the marginal log-likelihood $\log p_{\boldsymbol{\theta}}(x, y)$ is:

$$\begin{aligned} -\mathcal{L}(\boldsymbol{x}, y) =& \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}_A, \boldsymbol{z}_P|\boldsymbol{x}, y)}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}_A, \boldsymbol{z}_P, y) + \log p(y, \boldsymbol{z}_P, \boldsymbol{z}_A) - \log q_{\boldsymbol{\phi}}(\boldsymbol{z}_A, \boldsymbol{z}_P|\boldsymbol{x}, y)] \\ =& \underbrace{\mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}_A, \boldsymbol{z}_P|\boldsymbol{x}, y)}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}_A, \boldsymbol{z}_P, y) + \log p(y)]}_{(1)} - \underbrace{D_{KL}[p(\boldsymbol{z}_P, \boldsymbol{z}_A)||q_{\boldsymbol{\phi}}(\boldsymbol{z}_A, \boldsymbol{z}_P|\boldsymbol{x}, y)]}_{(2)} \\ \leq& \log p_{\boldsymbol{\theta}}(\boldsymbol{x}, y) \end{aligned} \tag{5.10}$$

As far as the prior on y does not depend on $\boldsymbol{\theta}$ neither on $\boldsymbol{\phi}$, the supervised loss is equivalent to a classic VAE loss as we have seen in Chapter 2. The loss is evaluated as follows: $\boldsymbol{x}$ and y are propagated through the encoder to compute the posterior distribution $q_{\boldsymbol{\phi}}(\boldsymbol{z}_A, \boldsymbol{z}_P|\boldsymbol{x}, y)$, from which K Monte Carlo samples are drawn. Then, the sampled latent variables $\boldsymbol{z}_P^{(k)}, \boldsymbol{z}_A^{(k)}$ are propagated through the decoder to compute the likelihoods $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}_A^{(k)}, \boldsymbol{z}_P^{(k)}, y)$ and to approximate the first term of equation 5.10. The Kullback-Leibler divergence, *i.e.* the second term of the equation, is evaluated analytically.

Second, when the label is not observed, the evidence lower bound of the marginal log-likelihood $\log p_{\boldsymbol{\theta}}(\boldsymbol{x})$ is:

$$\begin{aligned} -\mathcal{U}(\boldsymbol{x}) =& \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}_A, \boldsymbol{z}_P, y|\boldsymbol{x})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}_A, \boldsymbol{z}_P, y) + \log p_{\boldsymbol{\theta}}(y, \boldsymbol{z}_P, \boldsymbol{z}_A) - \log q_{\boldsymbol{\phi}}(\boldsymbol{z}_A, \boldsymbol{z}_P, y|\boldsymbol{x})] \\ =& \sum_y q_{\boldsymbol{\phi}}(y|\boldsymbol{x})(-\mathcal{L}(\boldsymbol{x}, y)) + H(q_{\boldsymbol{\phi}}(y|\boldsymbol{x})) \\ \leq& \log p_{\boldsymbol{\theta}}(\boldsymbol{x}) \end{aligned} \tag{5.11}$$

where $H(q_\phi(y|\boldsymbol{x}))$ denotes the entropy of y given $\boldsymbol{x}$. Maximizing the entropy of the classifier on the unlabeled data prevents the classifier to be too much confident on unlabeled samples. The first term of equation 5.11 is high when likely classes, according to $q_\phi(y|\boldsymbol{x})$, lead to a high evidence lower bound of the marginal likelihood, while classes that lead to a low evidence lower bound have a low probability according to $q_\phi(y|\boldsymbol{x})$.

The categorical predictive distribution $q_\phi(y|\boldsymbol{x})$ only contributes in the second objective function (5.11). To remedy this, we use the trick of [Kingma et al., 2014] which consists in adding a classification loss to the total objective function, so that $q_\phi(y|\boldsymbol{x})$ also learns from labeled data. The final objective function is defined as follows:

$$\mathcal{J} := \sum_{\boldsymbol{x},y\sim\hat{p}_{labeled-data}} [\mathcal{L}(\boldsymbol{x},y) - \alpha \log q_\phi(y|\boldsymbol{x})] + \sum_{\boldsymbol{x}\sim\hat{p}_{unlabeled-data}} \mathcal{U}(\boldsymbol{x}) \tag{5.12}$$

where $\hat{p}_{labeled-data}$ and $\hat{p}_{unlabeled-data}$ are the empirical labeled and unlabeled data distribution, respectively, and α is a hyper-parameter that controls the relative weight between generative and purely discriminative learning.

Gradient stopping. Our contribution to the semi-supervised optimization scheme is what we refer as gradient stopping. The machine learning model f_A generates data features from the categorical variable y and the continuous variable $\boldsymbol{z}_A$. Because f_A has very poor extrapolation capabilities, some inconsistent values of $(\boldsymbol{z}_A, \boldsymbol{z}_P, y)$ can lead to a good reconstruction of the training data. To mitigate this issue, we do not back-propagate the gradients with regard to f_A parameters when y is not observed.

3.1.3 Inference

At inference, [Kingma et al., 2014] uses the approximate predictive distribution $q_\phi(y|\boldsymbol{x})$ to make predictions. However, although the true predictive distribution $p_{\boldsymbol{\theta}}(y|\boldsymbol{x})$ is intractable, we can compute $\arg\max_y p_{\boldsymbol{\theta}}(y|\boldsymbol{x})$ if we are only interested in the class predictions. As a matter of fact, assuming that $p_{\boldsymbol{\theta}}(y)$ is uniform, we have from Bayes rule:

$$p_{\boldsymbol{\theta}}(y|\boldsymbol{x}) = \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}|y)p_{\boldsymbol{\theta}}(y)}{p_{\boldsymbol{\theta}}(\boldsymbol{x})} \propto p_{\boldsymbol{\theta}}(\boldsymbol{x}|y) \tag{5.13}$$

Moreover, denoting $[\boldsymbol{z}_A\ \boldsymbol{z}_P]$ as $\boldsymbol{z}$, which is independent from y, we can write that:

$$\begin{aligned} p_{\boldsymbol{\theta}}(\boldsymbol{x}|y) &= \int p_{\boldsymbol{\theta}}(\boldsymbol{x},\boldsymbol{z}|y)\, d\boldsymbol{z} = \int p_{\boldsymbol{\theta}}(\boldsymbol{x}|y,\boldsymbol{z})p_{\boldsymbol{\theta}}(\boldsymbol{z}|y)\, d\boldsymbol{z} \\ &= \int p_{\boldsymbol{\theta}}(\boldsymbol{x}|y,\boldsymbol{z})p_{\boldsymbol{\theta}}(\boldsymbol{z})\, d\boldsymbol{z} = \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{\theta}}(\boldsymbol{z})} p_{\boldsymbol{\theta}}(\boldsymbol{x}|y,\boldsymbol{z}) \end{aligned} \tag{5.14}$$

Thus, we can perform Monte Carlo sampling to estimate $p_{\boldsymbol{\theta}}(\boldsymbol{x}|y)$. In order to decrease the variance of the estimation, we can sample $\boldsymbol{z}$ from $q_\phi(\boldsymbol{z}|\boldsymbol{x})$, performing importance sampling as follows:

$$\begin{aligned} p_{\boldsymbol{\theta}}(\boldsymbol{x}|y) &= \mathbb{E}_{\boldsymbol{z}\sim q_\phi(\boldsymbol{z}|\boldsymbol{x})} \frac{p_{\boldsymbol{\theta}}(\boldsymbol{z})}{q_\phi(\boldsymbol{z}|\boldsymbol{x})} p_{\boldsymbol{\theta}}(\boldsymbol{x}|y,\boldsymbol{z}) \\ &= \mathbb{E}_{y^*\sim q_\phi(y|\boldsymbol{x})} \mathbb{E}_{\boldsymbol{z}\sim q_\phi(\boldsymbol{z}|\boldsymbol{x},y^*)} \frac{p_{\boldsymbol{\theta}}(\boldsymbol{z})}{q_\phi(\boldsymbol{z}|\boldsymbol{x})} p_{\boldsymbol{\theta}}(\boldsymbol{x}|y,\boldsymbol{z}) \end{aligned} \tag{5.15}$$

To our knowledge, this derivation of $\arg\max_y p_{\boldsymbol{\theta}}(y|\boldsymbol{x})$ has not yet been used in the context of semi-supervised VAEs. Besides, this derivation offers a convenient way to estimate the uncertainty over the inferred latent variables. We can easily compute the empirical standard deviation of the $\boldsymbol{z}^{(i)}$s where $\boldsymbol{z}^{(i)} \sim q_\phi(\boldsymbol{z}|\boldsymbol{x},y^*)$ and $y^* \sim q_\phi(y|\boldsymbol{x})$.

3.1.4 Related work

Hybrid modeling can be divided into physics-based losses [Chen et al., 2020; Raissi et al., 2019; Wang et al., 2021; Wei et al., 2020] and physics-based models, in which our method fits. While many works have leveraged GANs optimized with additional regularization objectives to constrain distributions that produce physically realistic samples [Subramaniam et al., 2020; Zheng et al., 2020], physics-based models have made more use of latent generative models to embed physical layers in their design. [Yildiz et al., 2019] and [Linial et al., 2021] introduced VAEs which latent variables were grounded to physical quantities (such as position and velocity) and governed by Ordinary Differential Equations (ODEs). More related to our work, [Aragon-Calvo and Carvajal, 2020] solve an inverse problem by using a conventional autoencoder in which they substitute the decoder by a fully physical model. They demonstrated that the architecture of the decoder induces a natural disentanglement of the latent space. This model is a particular case of the formalism of physics-integrated VAEs introduced by [Takeishi and Kalousis, 2021], that is the most closely related work to our methodology. Physics-integrated VAE [Takeishi and Kalousis, 2021] is, to our knowledge, the first general framework that combines physics models and conventional neural networks in a variational autoencoder. The key idea of [Takeishi and Kalousis, 2021] is that the combination of an imperfect physical model with a machine learning model gives more precise results than an imperfect physical model alone and better generalization performances than a machine learning model alone. More precisely, they showed that the deductive biases introduced by the physical model regularize the machine learning model, yielding more robust local optima. Our model falls in their formalism as far as ϕ-VAE latent space is partitioned into two types of data, z_A and z_P which are inputs of a machine learning model f_A and of a physics model f_P. The differences between ϕ-VAE and our model, though, are twofold:

Unsupervised VS semi-supervised setting. The first difference, which is minor in architecture, but significant in optimization, is that the latent variables of ϕ-VAE are never observed. In contrast, we partially observe the discrete variable y of our model during optimization, as we discussed in section 3.1.2.

Imperfect VS perfect physics model. There is a major difference of philosophy between ϕ-VAE and p^{3}VAE. ϕ-VAE tackles problems for which an imperfect physics model can approximate the data distribution. In this context, the machine learning part f_A of ϕ-VAE models the residual error of its imperfect physics model f_P. Intuitively, z_A *explains* the part of the observation x that was not *explained* by z_P. This has two practical consequences:

- Firstly, the latent variables z_P and z_A of ϕ-VAE are dependent conditioned on x:

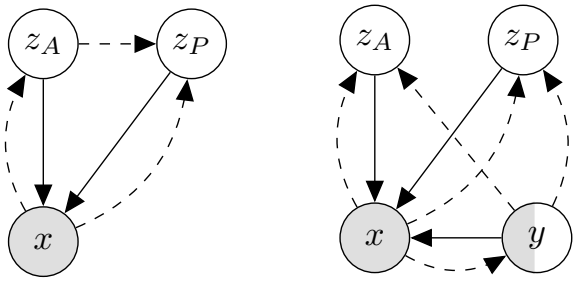
$$q_\phi(z_A, z_P|\boldsymbol{x}) := q_\phi(z_A|\boldsymbol{x})q_\phi(z_P|\boldsymbol{x}, z_A) \tag{5.16}$$

  In contrast, we assume that z_A and z_P are independent conditioned on x and y.
- Secondly, the expectation of the likelihood $p_{\boldsymbol{\theta}}(x|z_A, z_P)$ of ϕ-VAE is defined as follows:

$$p_\theta(x|z_P, z_A) := \mathcal{N}(x|\mathcal{F}[f_A, f_P; z_P, z_A], \Sigma) \tag{5.17}$$

  where $\mathcal{F}$ is a functional that evaluates $f_A \circ f_P$ or solves an equation in the form $f_A \circ f_P = 0$. The notable difference is that the machine learning model of ϕ-VAE is a function of the physics model output, which is the opposite for our model.

Because the physics part of ϕ-VAE is imperfect, a specific regularization technique is necessary to control the representation power of the machine learning part. In contrast, our experiments showed that neural networks in p^{3}VAE are less prone to impinge the physics model because they control different subspaces of the latent code.



(a) ϕ-VAE [Takeishi and Kalousis, 2021] (b) p^{3}VAE (ours)

Figure 5.8: Graphical model representations of (a) ϕ-VAE and (b) p^{3}VAE. Solid lines denote the generative model (a) $p_{\boldsymbol{\theta}}(x|z_A, z_P)$ and (b) $p_{\boldsymbol{\theta}}(x|z_A, z_P, y)$. Dashed lines denote the variational posterior approximation (a) $q_\phi(z_A|\boldsymbol{x})q_\phi(z_P|\boldsymbol{x}, z_A)$ and (b) $q_\phi(y|\boldsymbol{x})q_\phi(z_A, z_P|\boldsymbol{x}, y)$. White nodes are latent variables while gray ones are observed variables. Note that p^{3}VAE is, by nature, a semi-supervised model in contrast to ϕ-VAE which is unsupervised.

While some semi-supervised techniques use unlabeled data to regularize a machine learning model, hybrid modeling can be seen as a form of regularization that discard machine learning models that contradict the physics model.

Hybrid modeling has sometimes strong connections with disentanglement methods that aim to infer latent variables that are semantically meaningful. Intuitively, a latent space is disentangled when a change in a generative factor (*e.g.* irradiance) only affects one subset of the latent code (*e.g.* z_P). While some disentanglement techniques rely on the supervision (or semi-supervision) of the true generative factors of variations [Cheung et al., 2015; Karaletsos et al., 2015; Kulkarni et al., 2015; Rodriguez, 2021; Whitney et al., 2016], most recent methods tackle the problem of learning disentangled representations without supervision as far as true generative factors are usually not known in real applications. In that direction, several major works have extended the objective function of VAEs to foster disentanglement, such as [Higgins et al., 2017] that weights the Kullback-Leibler divergence term of the ELBO, [Chen et al., 2018; Kim and Mnih, 2018] that penalizes the total correlation, [Kumar et al., 2017] that regularize the expectation of the approximate posterior over observed data, or [Rezaabad and Vishwanath, 2020] that additionally maximize the mutual information between the latent variables and the observations. Meanwhile, Generative Antagonist Nets (GANs) [Goodfellow et al., 2014] can be used to learn interpretable representations. In particular, InfoGAN [Chen et al., 2016] is an information-theoretic extension of the GAN that demonstrated high disentanglement capabilities on complex data such as the CelebA face data set [Liu et al., 2015]. However, [Locatello et al., 2019] proved that the unsupervised learning of disentangled latent space is only possible with inductive biases on the learning approach and on the data. In that sense, they experimentally showed that the hyperparameters selection in order to reach high disentanglement does seem possible without the knowledge of true generative factors.

In the following, we show how can p^{3}VAE be applied to semantic segmentation of hyperpsectral data in section 3.2, and study how the physics-based deductive biases introduced in the architecture of the decoder can lead to a disentanglement of the latent space without additional regularization of the objective function in section 3.3.

3.2 Application of p^{3}VAE to hyperspectral image semantic segmentation

In this section, we apply p^{3}VAE to the semantic segmentation of high resolution hyperspectral images.

3.2.1 Latent variables and priors

The class of a pixel is modeled by the categorical variable $y \in \{1, \dots, C\}$. In what follows, we include *semantic* variability in *intrinsic* intra-class variability. In order to model the *intrinsic* and irradiance-based intra-class variabilities, we considered two latent variables, $\boldsymbol{z}_A$ and $\boldsymbol{z}_P$, respectively. Following the radiative transfer model we derived in section 2, we expect $\boldsymbol{z}_P$ to approximate δ_{dir}, $cos\,\Theta$, Ω and p_Θ. Although, inferring a four dimensional latent variable grounded to δ_{dir}, $cos\,\Theta$, Ω and p_Θ at the same time led to poor local optima, we constrained the latent space as follows:

$$\delta_{dir} cos\,\Theta \leftarrow z_P \tag{5.18}$$

$$\Omega\, p_\Theta \leftarrow g(z_P) \tag{5.19}$$

where g is an empirical function that will be discussed in section 3.3.4. Since $\delta_{dir} \in [0, 1]$ and $cos\,\Theta \in [0, 1]$, $\boldsymbol{z}_P$ should lie in $[0, 1]$. For this reason and because we would like a flexible distribution to model $\boldsymbol{z}_P$, we chose a Beta distribution.

The latent variable $\boldsymbol{z}_A$ is not grounded to a physical quantity. However, we could interpret $\boldsymbol{z}_A$ as the different reflectance spectra one class could take (*e.g.* different tree species). Thus, we modeled $\boldsymbol{z}_A$ with a Dirichlet distribution of dimension n_A which values can be interpreted as a distribution probability or as the abundance of different subclasses.

Prior distributions are defined as follows:

$$p_{\boldsymbol{\theta}}(y) := \mathcal{U}(\{1, \dots, C\}) \qquad p_{\boldsymbol{\theta}}(z_P) := Beta(z_P|\alpha^o, \beta^o) \qquad p_{\boldsymbol{\theta}}(\boldsymbol{z}_A) := Dir(\boldsymbol{z}_A|\gamma^o) \tag{5.20}$$

We empirically set α^o to 1 and $\beta^o = \frac{1-\cos\,\Theta^o}{\cos\,\Theta^o+\epsilon}\alpha^o$ where ϵ is a small constant to avoid division by zero and Θ^o is the solar incidence angle. This prior distribution favors high values of $\boldsymbol{z}_P$ while $\mathbb{E}_{p(\boldsymbol{z}_P)}\boldsymbol{z}_P = \cos\,\Theta^o$, which is what we expect on a flat ground. We set $\gamma^o = [1 \dots 1]_{1\times n_A}$, which is equivalent to having a uniform prior.

3.2.2 Encoder

The approximated posterior (parameterized by ϕ), *i.e.* the encoding part, is defined as follows:

$$q_{\boldsymbol{\phi}}(z_P, \boldsymbol{z}_A|\boldsymbol{x}, y) := q_{\boldsymbol{\phi}}(z_P|\boldsymbol{x}, y) q_{\boldsymbol{\phi}}(\boldsymbol{z}_A|\boldsymbol{x}, y) \tag{5.21}$$

where:

$$q_{\boldsymbol{\phi}}(z_P|\boldsymbol{x}, y) := Beta(z_P|\alpha(\boldsymbol{x}, y), \beta(\boldsymbol{x}, y)) \qquad q_{\boldsymbol{\phi}}(\boldsymbol{z}_A|\boldsymbol{x}, y) := Dir(\boldsymbol{z}_A|\gamma(\boldsymbol{x}, y)) \tag{5.22}$$

α, β and γ are neural networks that will be described in details in section 3.3.

3.2.3 Decoder

The decoder has a physical part f_P and a statistical part f_A, parameterized by θ. f_A estimates the true reflectance $\hat{\boldsymbol{\rho}} \in [0, 1]^B$, where B is the spectral dimension, based on the class y and the *intrinsic* variability z_A:

$$\begin{aligned} f_A :& \{1, \dots, C\} \times [0, 1]^{n_A} \longrightarrow [0, 1]^B \\ & (y, \boldsymbol{z}_A) \longmapsto \hat{\boldsymbol{\rho}} = \sum_{k=1}^{n_A} z_A[k] f_A^1(y)[k] \end{aligned} \tag{5.23}$$

where f_A^1 is a neural network. f_P is a deterministic function derived from the radiative transfer model described in section 2 that computes the observed reflectance $x \in [0,1]^B$ based on $\hat{\rho}$ and z_P:

$$\begin{aligned} f_P :&[0,1]^B \times [0,1] \longrightarrow [0,1]^B \\ &(\hat{\boldsymbol{\rho}}, z_P) \longmapsto \frac{z_P I^o_{dir} + g(z_P) I^o_{dif}}{cos\,\Theta^o \cdot I^o_{dir} + I^o_{dif}} \hat{\boldsymbol{\rho}} \end{aligned} \tag{5.24}$$

In contrast with our general framework, we do not model the decoding part of the VAE with a multivariate normal. Denoting $f_P(f_A(y, \boldsymbol{z}_A), z_P)$ as $\boldsymbol{\mu}$, we instead define the likelihood as follows:

$$p_{\boldsymbol{\theta}}(\boldsymbol{x}|y, z_P, \boldsymbol{z}_A) := \frac{1}{Z} \mathcal{N}(s|\mu, \sigma^2 I) \cdot exp(-\lambda\, arccos(\frac{\boldsymbol{x}^T \boldsymbol{\mu}}{\|x\|\|\mu\|})) \tag{5.25}$$

where σ and λ are hyperparameters and Z is a finite constant such that the density integrates to one. We demonstrate in the appendix that such a constant exists. The negative log-likelihood derives as follows:

$$-log\, p_{\boldsymbol{\theta}}(\boldsymbol{x}|y, z_P, \boldsymbol{z}_A) = \frac{1}{\sigma^2} MSE(\boldsymbol{x}, \boldsymbol{\mu}) + \lambda\, arccos(\frac{\boldsymbol{x}^T \boldsymbol{\mu}}{\|\boldsymbol{x}\|\|\boldsymbol{\mu}\|}) + C \tag{5.26}$$

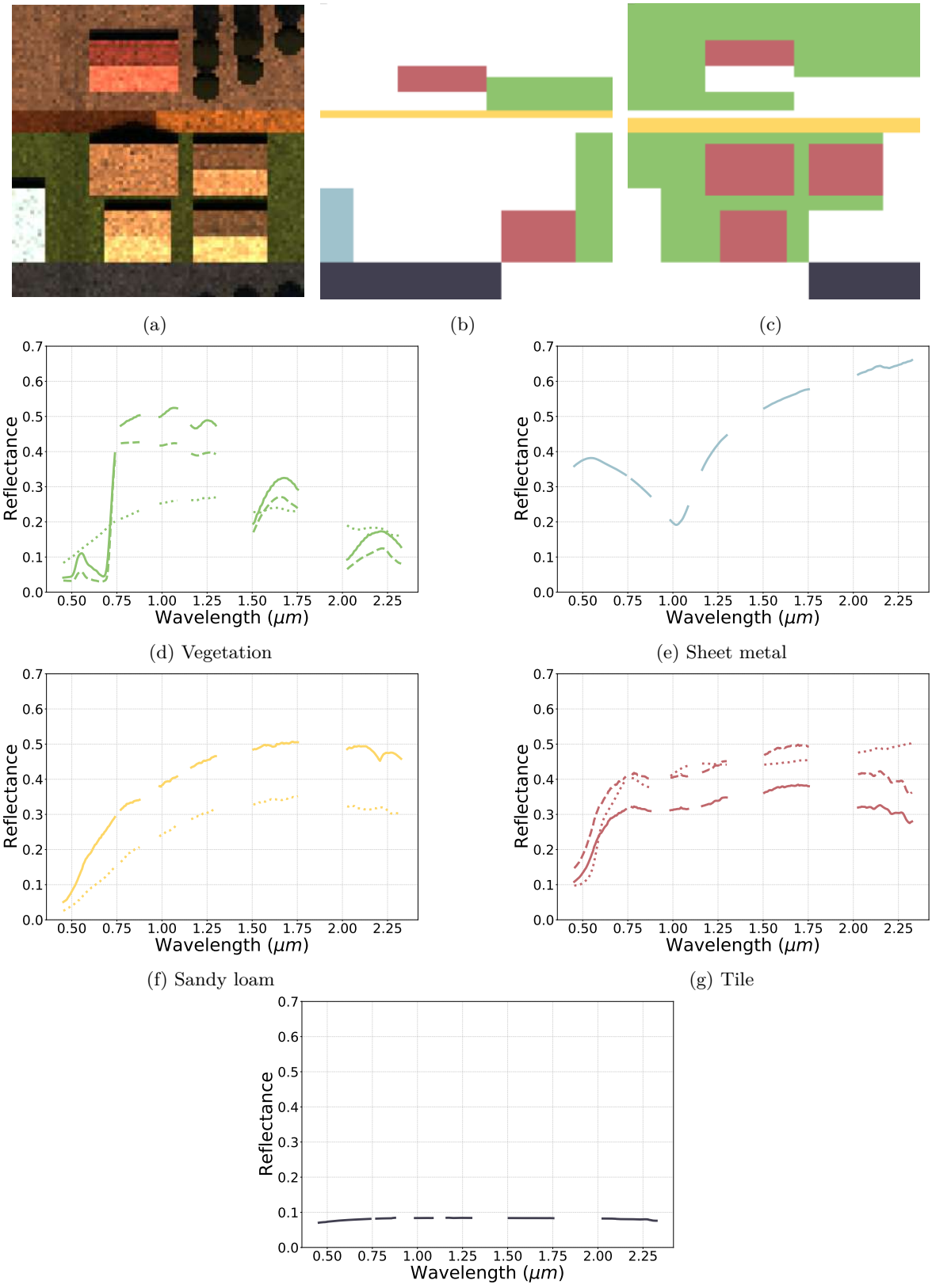
where C is a constant. Therefore, maximizing the likelihood of the data is equivalent to minimizing a linear combination of the mean squared error and the spectral angle between the observations $\boldsymbol{x}^{(i)}$ and $\boldsymbol{\mu}^{(i)} = f_P(f_A(y^{(i)}, \boldsymbol{z}_A^{(i)}), z_P^{(i)})$. The other consequence of this additional term in the density is that, without knowing the value of the constant Z, we cannot properly evaluate the likelihood for a given data point and neither sample from the likelihood. However, it is not an issue in our case because we are only interested in the predictions of the model.

3.3 Numerical experiments

3.3.1 Data

In order to evaluate the capacity of p^{3}VAE to handle spectral intra-class variability, we simulated hyperspectral data from several land cover classes with *intrinsic* (and *semantic*) intra-class variability under different irradiance conditions. Obviously, the interest of simulation is to have a precise knowledge of the radiative properties of the data, which will facilitate the quantitative evaluation of p^{3}VAE. Moreover, to assess the model performance on real data, we labeled a ground truth on the hyperspectral image of Toulouse, taking care to annotate pixels from the same semantic class with various irradiance conditions when possible. Both data sets are described hereafter.

Simulation of hyperspectral data under various irradiance conditions. We simulated an airborne hyperspectral image with the radiative transfer software DART [Gastellu-Etchegorry et al., 2012]. 300 spectral bands with a 6.5 nm resolution and a 0.5 m ground sampling distance were simulated, without simulating the Earth-atmosphere coupling. A false-color image and its ground truth are shown in Fig. 5.9. The scene was simulated with five materials (vegetation, sheet metal, sandy loam, tile and asphalt) whose reflectance spectra are shown in Fig. 5.9. Some classes gather different materials to have a realistic *intrinsic* intra-class variability. For instance, the vegetation class has three spectra which correspond to healthy grass, stressed grass and eucalyptus reflectances. We refer to those different materials within one class as subclasses. The azimuth angle is equal to 180 degrees and the solar zenithal angle is equal to 30 degrees with a nadir viewing. Besides, the roofs have different slopes. Moreover, to reproduce the scarcity of annotations in remote sensing, we labeled only a few pixels in the training data set and used the unlabeled pixels in our validation set. In particular, the training data set:



(a) (b) (c)

(d) Vegetation (e) Sheet metal

(f) Sandy loam (g) Tile

(h) Asphalt

Figure 5.9: (a) False color composition of the simulated hyperspectral image and its (b) training and (c) validation ground truths. Reflectance spectra of the land cover classes in the simulated scene in (d-e-f-g-h).

- does not include vegetation, asphalt and sandy loam pixels in the shadows,
- partially includes tile pixels on different slopes,
- contains all pixels of sheet metal.

Simulating a scene was very convenient as we precisely knew the topography and the reflectance of each material. To test our model, we computed roughly 10,000 test spectra per class with different slopes, direct irradiance and sky viewing angle. We also made different combinations of reflectance spectra within each class. Precisely, the factors that were used to generate a spectra $\boldsymbol{x}$ were the class y, the portion of direct irradiance δ_{dir}, the solar incidence angle Θ, the sky viewing angle factor Ω, the anisotropy correction coefficient p_Θ, the intra-class mixing coefficient $\alpha \in [0, 1]$ and the subclass configuration η:

$$\boldsymbol{x} = \frac{\delta_{dir} cos\ \Theta I^o_{dir} + \Omega\ p_\Theta I^o_{dif}}{cos\ \Theta^o \cdot I^o_{dir} + I^o_{dif}} \tilde{\boldsymbol{\rho}} \tag{5.27}$$

with:

$$\tilde{\boldsymbol{\rho}} = \alpha \boldsymbol{\rho}[y][\eta_1] + (1 - \alpha)\boldsymbol{\rho}[y][\eta_2] \tag{5.28}$$

where $\boldsymbol{\rho}[y][\eta_i]$ denotes the reflectance spectrum of the η_ith subclass of class y. For the test data set, the simulation of spectra rather than images was very convenient to cover the whole range of variation factors.

3.3.2 Annotation of a real hyperspectral image

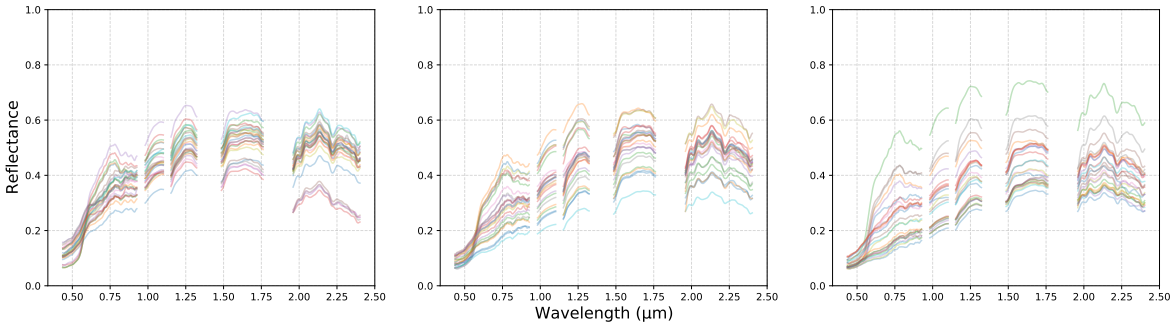
We used subsets of the recently published airborne hyperspectral data set of the CAMCATT-AI4GEO experiment[1] in Toulouse, France [Roupioz et al., 2023]. Data was acquired with a AisaFENIX 1K camera, which covers the spectral range from 0.4 µm to 2.5 µm with a 1 m ground sampling distance. Data was converted in radiance at aircraft level through radiometric and geometric corrections. Then, the radiance image was converted to surface reflectance with the atmospheric correction algorithm COCHISE [Miesch et al., 2005] (that makes a flat surface assumption). We labeled a ground truth through a field campaign and photo interpretation on various areas in Toulouse. We selected eight land cover classes (Tile, Asphalt, Vegetation, Painted sheet metal, Water, Gravels, Metal and Fiber cement) which were spatially split in a labeled training set, an unlabeled training set and a test set (spectra are shown on Fig. 5.10). The labeled training set, unlabeled training set and test set contain 3671 pixels, 7762 pixels and 10333 pixels, respectively. We also selected three areas to qualitatively evaluate the quality of the land cover maps, shown in Fig. 5.14.

3.3.3 Models architectures

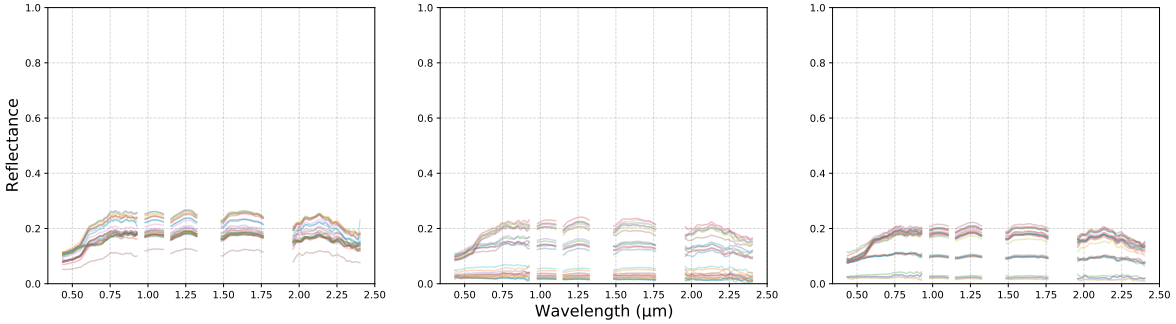
We compared our model, p^{3}VAE, with a conventional CNN, a conventional semi-supervised gaussian VAE [Kingma et al., 2014], a semi-supervised InfoGAN [Spurr et al., 2017] and a physics-guided VAE, a variant of p^{3}VAE. On real data, we also tested a semi-supervised version of FG-UNET [Stoian et al., 2019], a deep FCN designed for the semantic segmentation of remote sensing images. In this section, we present in details the architecture of the models, the optimization details, the metrics and the results.

p^{3}VAE. As seen in section 3.2, the parameters of the posterior distributions $q_\phi(z_P|\boldsymbol{x}, y)$ and $q_\phi(\boldsymbol{z}_A|\boldsymbol{x}, y)$ are computed with neural networks α, β and γ. α and β are dense neural

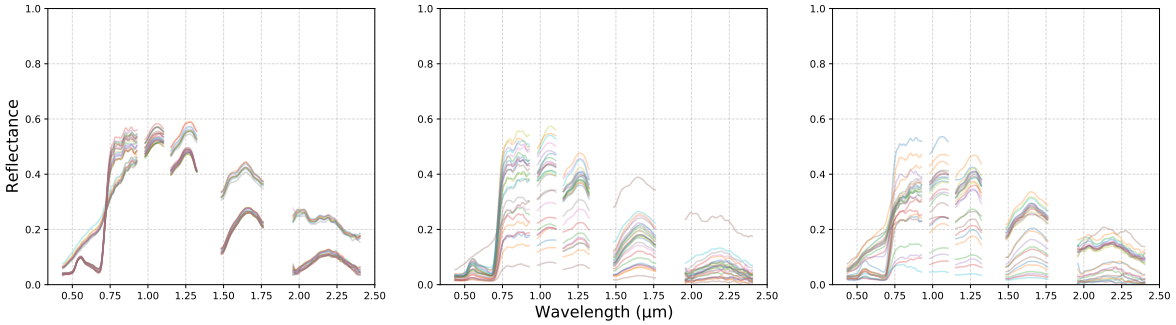
[1]Data is publicly available here: https://camcatt.sedoo.fr/



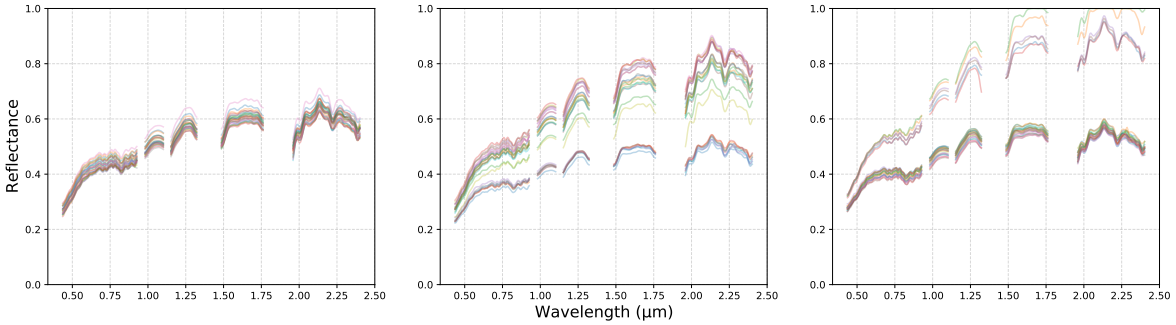
(a) Tile



(b) Asphalt

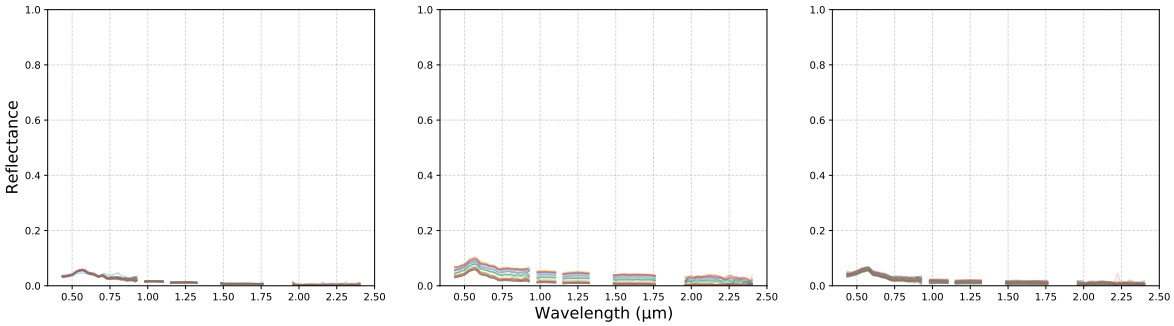


(c) Vegetation

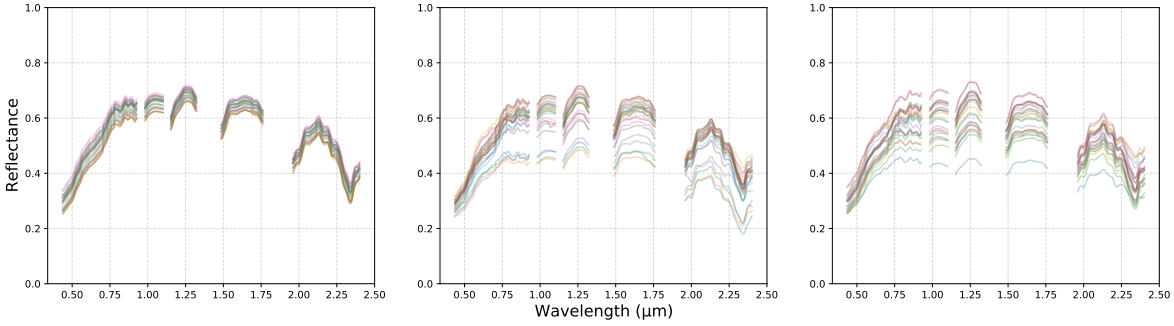


(d) Painted sheet metal

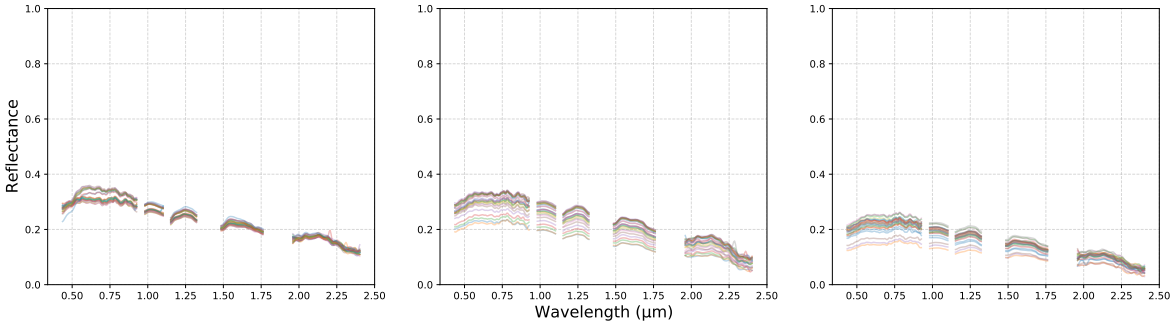
Figure 5.10: Spectra from the (left) labeled, (middle) unlabeled and (right) test real data sets.



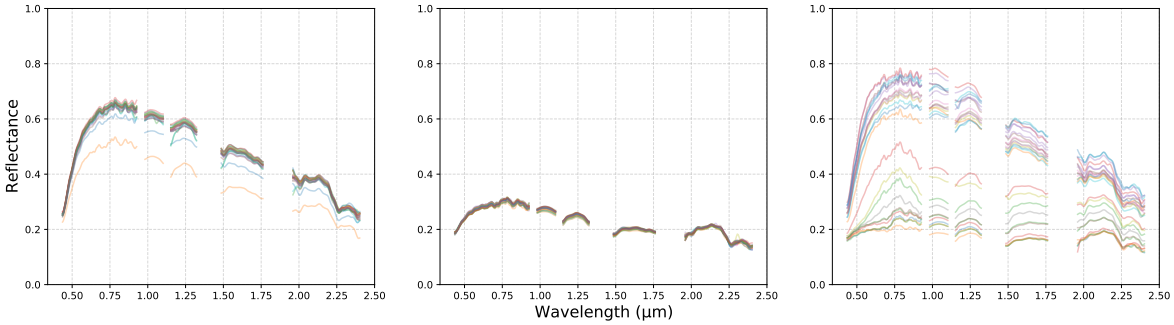
(e) Water



(f) Gravels



(g) Metal



(h) Fiber cement

Figure 5.10: Spectra from the (left) labeled, (middle) unlabeled and (right) test real data sets.

networks because we believe that z_P can be inferred only with affine transformations of $\boldsymbol{x}$ and y. As far as $\boldsymbol{z}_A$ is more related to the shape of $\boldsymbol{x}$, we found more relevant to model γ by a convolutional neural network. Concerning the decoder, f_A^1 is the element-wise multiplication of two $B \times n_A$ matrices computed with dense neural networks. The approximate predictive distribution $q_\phi(y|\boldsymbol{x})$ is computed with a convolutional neural network that has rigorously the same architecture than the CNN model described below. The model is trained using gradient stopping as described in section 3.1.

CNN. The CNN architecture is very similar to the CNN introduced in [Hu et al., 2015] that is a common architecture in the hyperspectral literature [Audebert et al., 2019]. It is composed of two spectral convolutions (with one kernel per continuous spectral domain), a skip connection and a max-pooling layer, followed by fully connected layers.

Gaussian VAE. We used a conventional semi-supervised gaussian VAE [Kingma et al., 2014]. The encoder of the gausian VAE uses the same convolutional neural network used in the encoder of p^{3}VAE. The dimension of the latent variable z is equal to $\dim(\boldsymbol{z}_A) + \dim(z_P)$. The decoder is composed of dense layers and a final sigmoid activation.

ssInfoGAN. We followed the technique introduced in [Spurr et al., 2017] to guide an InfoGAN model [Chen et al., 2016] with semi-supervision. The generator and the discriminator head of the semi-supervised InfoGAN are dense neural networks. The generator takes inputs of size $\dim(\boldsymbol{z}_A) + \dim(z_P) + \dim(\boldsymbol{z})$, where $\boldsymbol{z}$ is the incompressible noise (see [Chen et al., 2016] for exhaustive details on the InfoGAN model). The neural network that computes the auxiliary distribution for the categorical latent code has the same architecture than the dense layers in the CNN above. The neural network that computes the auxiliary distribution for the continuous latent code is a dense neural network.

FG-UNET. FG-UNET [Stoian et al., 2019] enhances a classical U-net [Ronneberger et al., 2015] for remote sensing images. In particular, a pixel-wise fully connected path is added to improve the geometric consistency of the predictions to deal with sparse annotations [Stoian et al., 2019]. The encoding part of FG-UNET is composed of five spatial-spectral convolutional layers with two max-pooling operations in-between. The decoding part is composed of five transposed convolutional layers with two up-sampling operations in-between.

Physics-guided VAE. In order to compare the benefits of the physics model only, we designed a physics-guided VAE which encoder and decoder have the same architectures that the ones of p^{3}VAE. The only difference is that the decoder has only a machine learning part f_A. The model is trained using gradient stopping as described in section 3.1.

Every classifiers are turned into bayesian classifiers using MC Dropout [Gal and Ghahramani, 2016]. Fig. 5.17 illustrates the architectures of the different models. Layers in **pink**, **purple**, **light purple** and **green** denote input data, output of a convolutional layer, output of a fully-connected layer, and output of a pooling layer, respectively. Other uncommon operations are specifically described in the figure.

3.3.4 Optimization

The CNN and the VAE-like models were optimized with the Adam stochastic descent gradient algorithm [Kingma and Ba, 2015] for 100 and 50 epochs on the simulated and real data, respectively, and a batch size of 64. We optimized ssInfoGAN in the Wasserstein training fashion [Arjovsky et al., 2017] with the gradient penalty loss introduced in [Gulrajani et al., 2017], using the RMSProp algorithm for 200 and 150 epochs on the simulated and real

Table 5.1: Mean F1 score per class over 10 runs

Inference model		**Classes**					
		Vegetation	Sheet metal	Sandy loam	Tile	Asphalt	Average
CNN	$q_{\phi}(y\|\boldsymbol{x})$	0.90	0.81	0.77	0.79	0.75	0.80
	$q_{\phi}(y\|\boldsymbol{x})$ (full annotations)[1]	0.92	0.79	*0.90*	0.87	0.86	0.86
ssInfoGAN	$q_{\phi}(y\|\boldsymbol{x})$	0.86	0.79	0.75	0.75	0.69	0.77
Gaussian VAE	$q_{\phi}(y\|\boldsymbol{x})$	0.93	0.80	0.87	0.86	0.74	0.84
	$\arg\max_y p_{\boldsymbol{\theta}}(y\|\boldsymbol{x})$	0.94	0.88	**0.90**	**0.92**	0.85	0.90
Physics-guided VAE	$q_{\phi}(y\|\boldsymbol{x})$	0.93	0.81	0.86	0.86	0.76	0.84
	$\arg\max_y p_{\boldsymbol{\theta}}(y\|\boldsymbol{x})$	0.86	0.85	0.83	0.75	0.77	0.81
p^{3}VAE	$q_{\phi}(y\|\boldsymbol{x})$	0.92	0.82	0.88	0.87	0.81	0.86
	$\arg\max_y p_{\boldsymbol{\theta}}(y\|\boldsymbol{x})$	**0.96**	**0.97**	**0.90**	0.90	**0.93**	**0.93**

[a]The CNN was optimized with every pixels labeled on the image (*i.e.* the labeled and unlabeled sets showed on Fig. 5.9), in contrast with every other cases where the image is partially labeled.

data, respectively, and a batch size of 64. FG-UNET was trained with the Adam algorithm for 500 epochs and a batch size of 16. In [Stoian et al., 2019], FG-UNET is optimized by minimizing a standard cross-entropy between labels and predictions. In our experiments, we add an unsupervised reconstruction loss on the L1 norm like in [Castillo-Navarro et al., 2021] to guide the training of FG-UNET with unlabeled pixels. We tuned the learning rate between $5 \cdot 10^{-5}$ and $1 \cdot 10^{-4}$ to reach loss convergence on the training and validation sets. We modeled p^{3}VAE empirical function $g : z_P \longmapsto \Omega\, p_{\Theta}$ by an affine transformation and tuned its parameters on the validation set. We retained $g(z_P) = z_P + 0.2$, which led to good spectral reconstruction under low direct irradiance conditions despite being very simplistic. We weighted the Kullback-Leibler divergence term in equations (5.10) and (5.11) by $\beta = 10^{-4}$. We also weighted the entropy term in equation (5.11) to balance between high accuracy and high uncertainty with a 10^{-1} coefficient. Besides, we applied a Ridge regularization on the weights of the classifiers and encoders with a 10^{-2} penalty coefficient. All hyperparameters were selected through random search.

3.3.5 Results - simulated data

In the experiments on the simulated data, we reported the F1 score and the JEMMIG score defined in chapter 3, as well as the local irradiance estimate against the true irradiance and the estimated reflectance against the true reflectance.

Accuracy metrics

F1 score. Mean F1 score per class over 10 runs is shown on table 5.1. First, all semi-supervised models outperformed the fully supervised CNN in terms of average F1 score by a margin of at least 4%. Second, using $q_{\phi}(y|\boldsymbol{x})$ to make the predictions, semi-supervised models had small differences in term of F1 score, except for ssInfoGAN. Third, making predictions using $\arg\max_y p_{\boldsymbol{\theta}}(y|\boldsymbol{x})$, large gains were obtained with the gaussian VAE and p^{3}VAE over the conventional CNN. In particular, p^{3}VAE made significant improvements over the gaussian VAE (+3%), the physics-guided VAE (+9%) and the CNN (+13%). A statistical hypothesis test shows that the average F1 score of the gaussian VAE and p^{3}VAE are significantly different: we reject the hypothesis of equal average with a 0.3% p-value. Even with exhaustive annotations (we added the labels of the "unlabeled" pixels in the training data set and we refer to this setting as *full annotations* in table 5.1), p^{3}VAE outperformed the CNN by a 7% margin. Besides, we can notice that the accuracy for *Sheet metal*, which

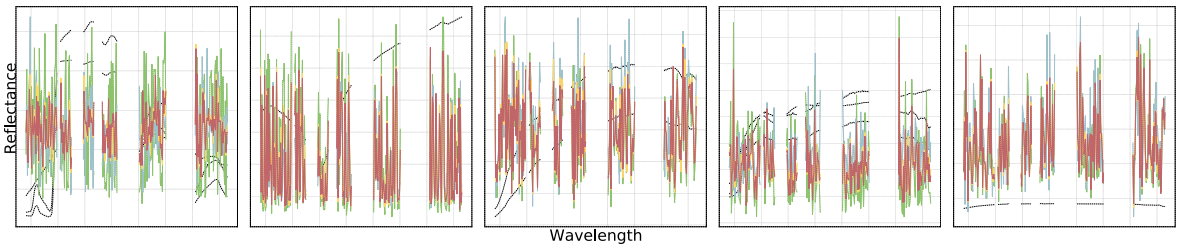
Table 5.2: Mean JEMMIG metric over 10 runs

Metric	Model	Factors					Average
		y	$\delta_{dir} cos\,\Theta$	Ωp_Θ	α	η	
Joint entropy (↓)[1]	Gaussian VAE	2.8	6.9	6.9	4.5	4.5	5.1
	Physics-guided VAE	3.1	6.4	6.0	**4.4**	**4.4**	4.9
	ssInfoGAN	3.2	**6.3**	**5.4**	4.5	4.5	**4.8**
	p^{3}VAE	**2.7**	7.3	7.2	**4.4**	**4.4**	5.2
Mutual information gap (↑)[1]	Gaussian VAE	0.62	0.18	**0.096**	0.20	0.20	0.26
	Physics-guided VAE	1.0	0.068	0.021	0.28	0.28	0.33
	ssInfoGAN	0.92	0.16	0.040	0.29	0.29	0.34
	p^{3}VAE	**1.2**	**0.85**	0.012	**0.31**	**0.31**	**0.54**
Normalized JEMMIG score (↑)[1]	Gaussian VAE	0.67	0.22	0.12	0.39	0.39	0.36
	Physics-guided VAE	0.69	0.26	0.21	**0.42**	**0.42**	0.40
	ssInfoGAN	0.66	**0.28**	**0.29**	0.41	0.41	**0.41**
	p^{3}VAE	**0.78**	0.25	0.061	**0.42**	**0.42**	0.38

[a](↓) means that the lower is the better while (↑) means that the higher is the better.

is the only class with homogeneous irradiance on the training image (while, thet test data covers all irradiance configurations), is barely the same for every models when $q_{\boldsymbol{\phi}}(y|\boldsymbol{x})$ is used. Finally, better predictions were made when computing $\arg\max_y p_{\boldsymbol{\theta}}(y|\boldsymbol{x})$ except for the physics-guided VAE while ssInfoGAN performed poorly with a 3% lower F1 score than the CNN. p^{3}VAE reached a 0.97 F1 score for the class *Sheet metal*, outperforming other methods by a 9% margin at least.

Local irradiance estimate. Fig. 5.11 shows, for each class, the inferred z_P by p^{3}VAE against the true $\delta_{dir} cos\,\Theta$ of the test data set. Correctly classified pixels are shown as purple points while wrongly predicted pixels are shown as orange points. The size of the points is proportional to the exponential of the empirical standard deviation of z_P. First, we notice that most confusions are made when $\delta_{dir} cos\,\Theta$ is poorly estimated or when its true value is low. Secondly, we see that most confusions go hand in hand with high z_P uncertainty estimates, meaning that two classes are likely, but under very different irradiance conditions. Finally, we see that there is a bias, in the sense that the average prediction of $\delta_{dir} cos\,\Theta$ is different from its true value. $\delta_{dir} cos\,\Theta$ is mostly under-estimated, which is the counterpart of over-estimated reflectance spectra as shown in Fig. 5.11 and commented below.



(a) (b) (c) (d) (e)

Figure 5.12: Estimated class reflectance for (a) vegetation, (b) sheet metal, (c) sandy loam, (d) tile and (e) asphalt with the physics-guided VAE. True reflectance spectra, that were used for data simulation, are plotted in dashed black lines.

Estimated reflectance. Fig. 5.11 shows the estimated reflectance spectra by p^{3}VAE. There are four estimated spectra per class as we set $n_A = 4$ in our experiments. p^{3}VAE rather well estimated the shape of the spectra. For instance, the absorption peak of clay at 2.2 µm is reconstructed by p^{3}VAE for the classes Tile and Sandy loam. However, the intensity of the

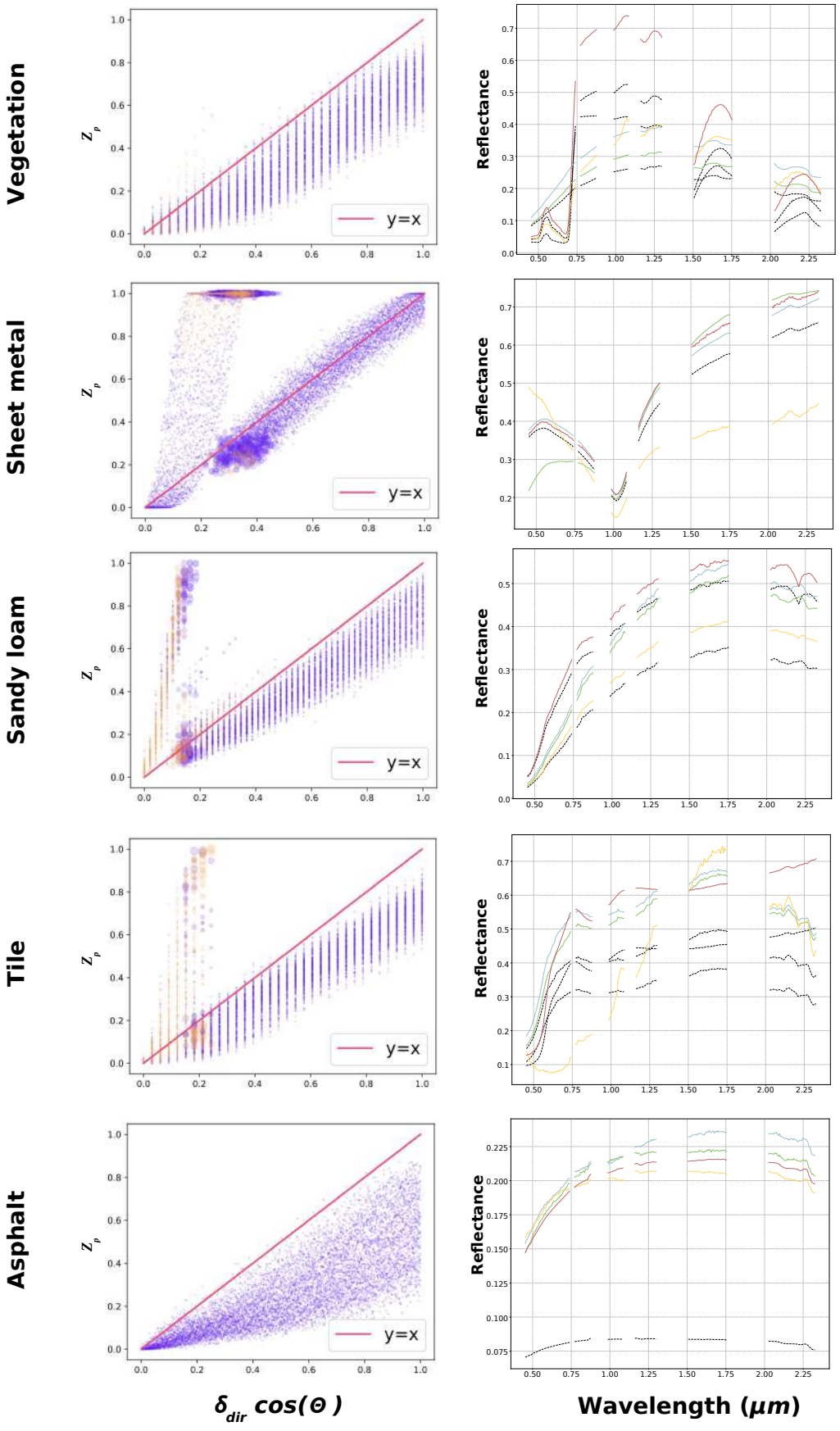


Figure 5.11: **On the left column**: predicted z_P against true $\delta_{dir} cos\,\Theta$ for each class with p^{3}VAE. Each point represents a pixel, which is correctly classified if shown in purple, or wrongly classified if shown in orange. The size of the points is proportional to the exponential of the empirical standard deviation of z_P. **On the right column**: Estimated class reflectance for each class with p^{3}VAE. True reflectance spectra, that were used for data simulation, are plotted in dashed black lines.

spectra is not accurate, which is a consequence of the bias in the $\delta_{dir} cos\,\Theta$ prediction. Finally, we highlight that p^{3}VAE inferred realistic reflectance spectra whereas the estimated spectra by the physics-guided VAE look like white noise, as shown in Fig. 5.12.

■ **Disentanglement**

JEMMIG. Table 5.2 shows the averaged JEMMIG metric for each factor over 10 runs. y is the class of the pixel, $\delta_{dir} cos\,\Theta$ and Ωp_Θ are the local irradiance conditions, α is the intra-class mixing coefficient and η represents the subclasses configuration (as described in section 3.3.2). First, the best overall joint entropy was reached by ssInfoGAN. Differences of joint entropy for the factors α and η are not significant though and the best joint entropy for the class factor was achieved by p^{3}VAE (a statistical hypothesis test shows that p^{3}VAE reached a significantly lower joint entropy than the gaussian VAE with a 0.9% p-value). The same observations can be drawn on the normalized JEMMIG score. In contrast, p^{3}VAE outperformed other models in term of mutual information gap (MIG), except for the Ωp_Θ factor. Largest discrepancies between the models in term of MIG were observed for the irradiance conditions factors. The best MIG score for the $\delta_{dir} cos\,\Theta$ factor, reached by p^{3}VAE, was more than twice the score reached by ssInfoGAN. In contrast, the MIG for Ωp_Θ was 8 times lower with p^{3}VAE than the gaussian VAE, although every MIG scores for this factor were near zero.

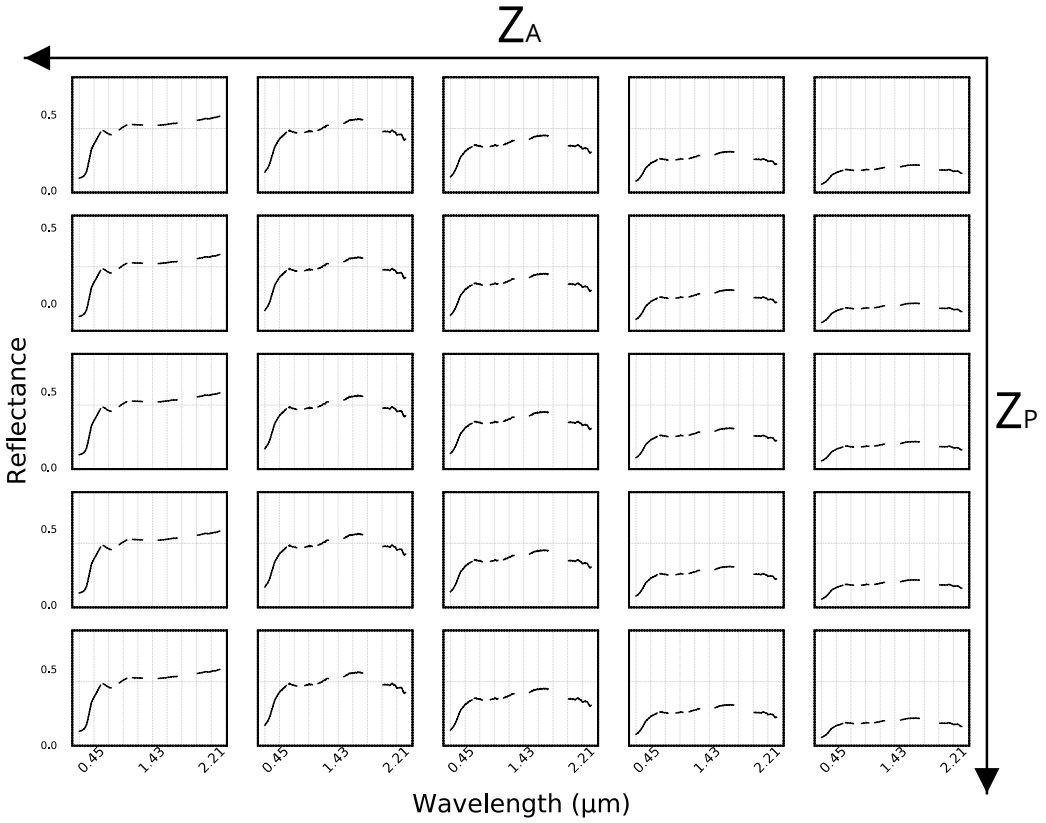
Maximum likelihood estimates. A qualitative way to evaluate the disentanglement of the latent space is to plot spectra that maximize the generative model likelihood for different values of the latent variables. Fig. 5.13 shows the maximum likelihood estimates of the class tile for the physics-guided VAE, p^{3}VAE, the gaussian VAE and ssInfoGAN. The different estimates were obtained for different realizations of $\boldsymbol{z}_A$ and z_P. z_P was interpolated between its lowest and highest values inferred on the training set. $\boldsymbol{z}_A$ was interpolated between two values inferred on different types of tile under high and low direct irradiance conditions. We see on Fig. 5.13 that latent variables z_P and $\boldsymbol{z}_A$ are disentangled by p^{3}VAE. Firstly, for a given column, the variations of the spectra are only induced by the variation in irradiance conditions : the shape roughly stays the same but the intensity changes. Secondly, for a given row, the variations of the spectra are only induced by the change in the nature of matter : the intensity is the same but the shape changes. In contrast, the physics-guided VAE does not disentangle the latent variables. A variation along one latent variables induces a change both in the irradiance conditions and on the type of tile.

3.3.6 Results - real data

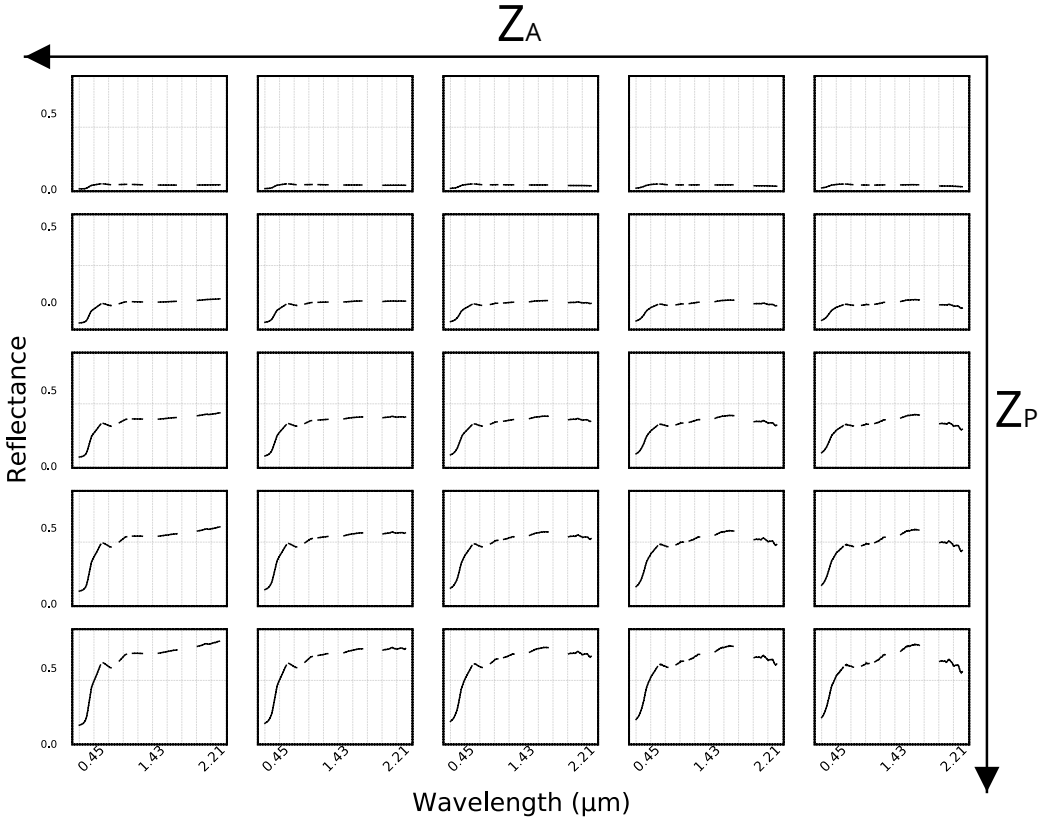
■ **Accuracy metrics**

F1 score. Mean F1 score per class over 10 runs is showed on table 5.3. The CNN with exhaustive annotations outperformed every models, including p^{3}VAE by a small margin of 1%. Excluding the CNN with exhaustive annotations, p^{3}VAE reached a better F1 score than other models (11%, 8%, 16%, 10% and 2% higher than the CNN, FG-UNET, ssInfoGAN, the gaussian VAE and the physics-guided VAE, respectively, using $q_{\boldsymbol{\phi}}(y|\boldsymbol{x})$). In contrast with the experiments on the simulated data set, VAE-like models obtained better performances using the predictive distribution $q_{\boldsymbol{\phi}}(y|\boldsymbol{x})$ rather than $\arg\max_y p_{\boldsymbol{\theta}}(y|\boldsymbol{x})$.

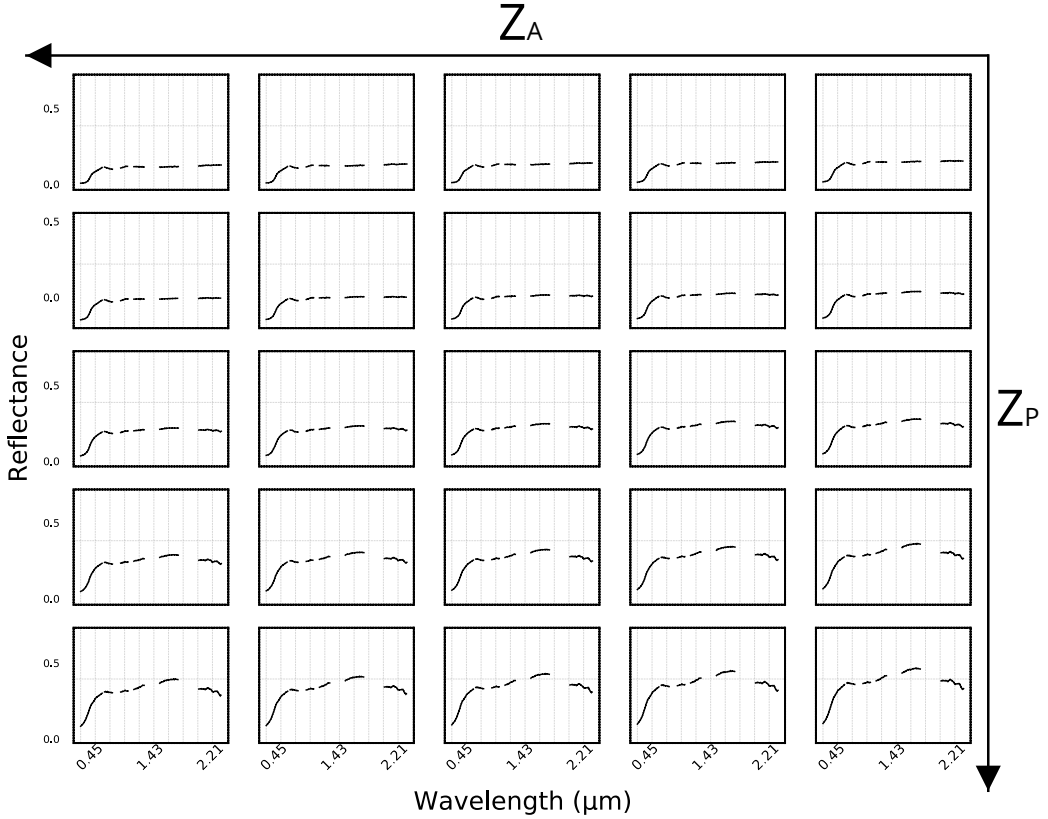
Land cover maps. Fig. 5.14 shows land cover maps produced by the CNN and p^{3}VAE on three test scenes that allow to qualitatively assess the accuracy of the models. We do not have a labeled ground truth of those scenes, which moreover include pixels belonging to classes not represented in the training data set. However, we focus on specific areas for



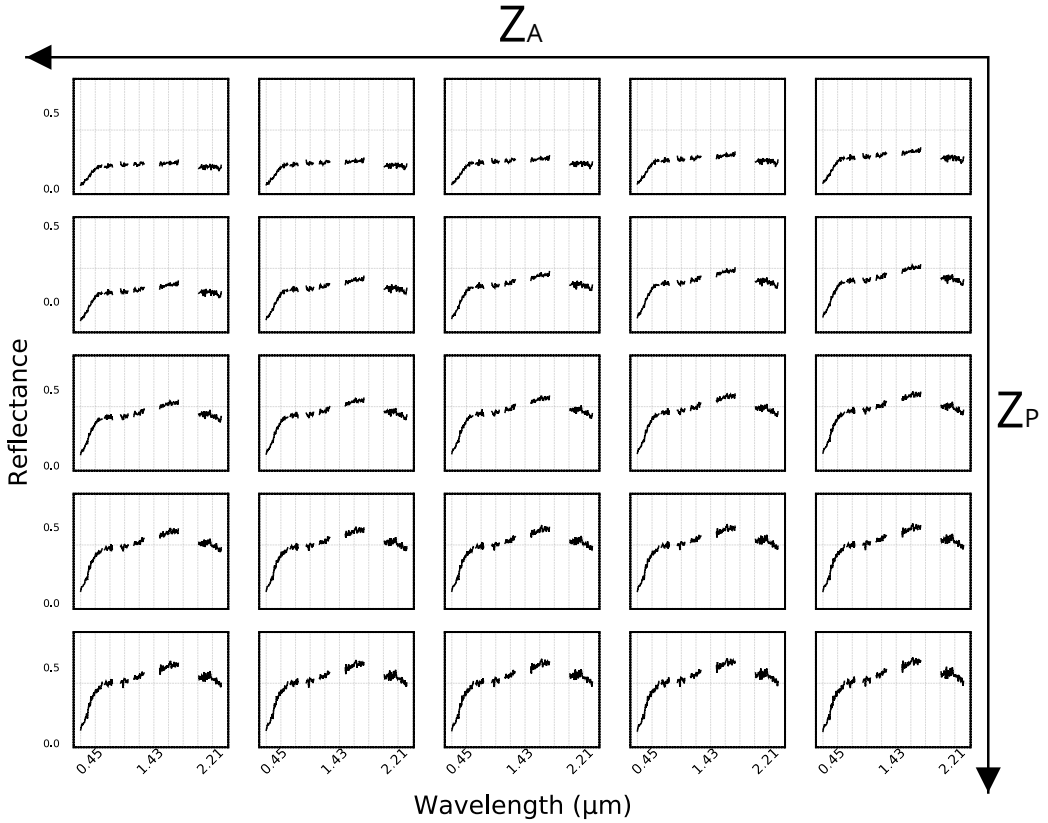
(a) Physics-guided VAE



(b) p^{3}VAE



(c) Gaussian VAE



(d) ssInfoGAN

Figure 5.13: Maximum likelihood estimates of (a) the physics-guided VAE, (b) p^{3}VAE, (c) the gaussian VAE and (d) ssInfoGAN, along z_P (on the y-axis) and $\boldsymbol{z}_A$ (on the x-axis) for the class *Tile*. z_P is interpolated between its minimum (top row) and maximum (bottom row) values inferred on the training data set. $\boldsymbol{z}_A$ is interpolated between two realizations of $\boldsymbol{z}_A$ that were inferred on two different types of tile of the training set (left and right columns) under high and low direct irradiance conditions, respectively.

Table 5.3: Mean F1 score per class over 10 runs

Inference model		Classes[1] #1	#2	#3	#4	#5	#6	#7	#8	Avg
CNN	$q_\phi(y\|\boldsymbol{x})$	0.92	0.41	0.91	0.92	0.83	0.89	0.87	0.56	0.79
	$q_\phi(y\|\boldsymbol{x})$ (full annotations)[2]	*0.96*	*0.53*	*0.98*	*1.00*	*0.99*	0.98	*0.99*	*0.87*	*0.91*
FG-Unet	$q_\phi(y\|\boldsymbol{x})$	0.94	0.50	0.87	0.97	0.54	0.98	0.92	**0.83**	0.82
ssInfoGAN	$q_\phi(y\|\boldsymbol{x})$	0.86	0.34	0.78	0.95	0.40	0.94	0.91	0.76	0.74
Gaussian VAE	$q_\phi(y\|\boldsymbol{x})$	0.94	0.27	0.88	0.96	0.83	0.92	**0.96**	0.66	0.80
	$\arg\max_y p_{\boldsymbol{\theta}}(y\|\boldsymbol{x})$	0.82	0.40	0.86	0.73	0.84	0.60	0.79	0.63	0.71
Physics-guided VAE	$q_\phi(y\|\boldsymbol{x})$	**0.95**	0.48	0.97	**1.00**	0.98	0.98	0.87	0.78	0.88
	$\arg\max_y p_{\boldsymbol{\theta}}(y\|\boldsymbol{x})$	0.92	0.48	0.97	0.99	**1.00**	**0.99**	0.84	0.78	0.87
p^{3}VAE	$q_\phi(y\|\boldsymbol{x})$	**0.95**	**0.51**	**0.98**	0.99	0.98	**0.99**	**0.96**	0.81	**0.90**
	$\arg\max_y p_{\boldsymbol{\theta}}(y\|\boldsymbol{x})$	0.93	0.49	0.97	0.91	0.99	0.79	0.90	0.78	0.84

[1] #1: Tile, #2: Asphalt, #3: Vegetation, #4: Painted sheet metal, #5: Water, #6: Gravels, #7: Metal, #8: Fiber cement. [2] The CNN was optimized with every pixels labeled on the image (*i.e.* the labeled and unlabeled sets), in contrast with every other cases where the image is partially labeled.

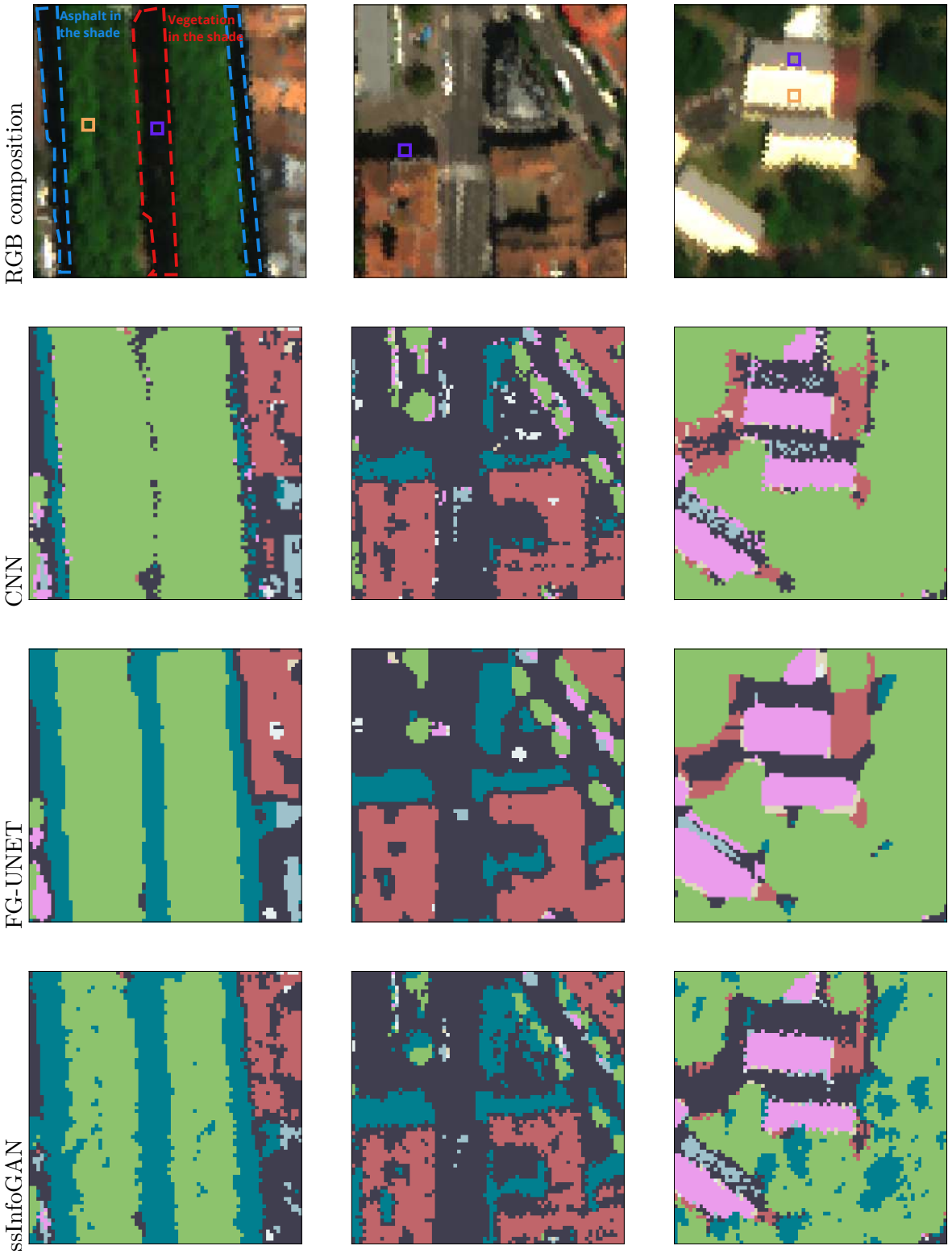
which we know the land cover with confidence. On the first scene, the ground vegetation in the shade between the rows of trees was partially confused with asphalt by the CNN, while p^{3}VAE was correct. On the first and the second scenes, asphalt in the shade of buildings was also partially confused with water by the CNN, while p^{3}VAE was correct. On the third scene, the least lit fiber cement roofs were mainly confused with asphalt and metal by the CNN while p^{3}VAE made much fewer confusions. Finally, the advantage of FG-UNET with respect to the CNN mainly lies in its spatial regularization property. The land cover maps produced by FG-UNET are much more geometrically consistent than the land cover maps of the spectral models.

■ **Disentanglement**

Qualitative assessment of z_P interpretability. We can observe on Fig. 5.14 the predicted latent variable z_P over the scenes. Relative differences of z_P values within one material under different irradiance conditions are consistent. For instance, pixels on roofs that faces the sun (under high direct irradiance) have high values of z_P while pixels on roofs in the other direction (under low direct irradiance) have low values of z_P. However, we also observe that different materials with the same irradiance conditions can have different z_P values. For instance, pixels on the road and pixels on flat roofs (that roughly have the same direct irradiance) have different z_P values.

3.3.7 Ablation study

In this section, we study the impact of gradient stopping (described in section 3.1) on the optimization of p^{3}VAE. On simulated data, Table 5.4 shows that gradient stopping has not a significant influence on the model performance (confirmed by statistical hypothesis tests), for both inference techniques. However, a look at the estimated reflectance spectra by p^{3}VAE without gradient stopping shows that the machine learning part learns unrealistic spectra. For instance, we plot the estimated reflectance spectra of the class asphalt by p^{3}VAE without gradient stopping on Fig. 5.15. On real data, statistical hypothesis tests show that gradient stopping does have a significant influence on the model performance, for both inference techniques (with a 1.8% p-value for the inference with $q_\phi(y|\boldsymbol{x})$).


RGB composition
Asphalt in the shade
Vegetation in the shade
CNN
FG-UNET
ssInfoGAN

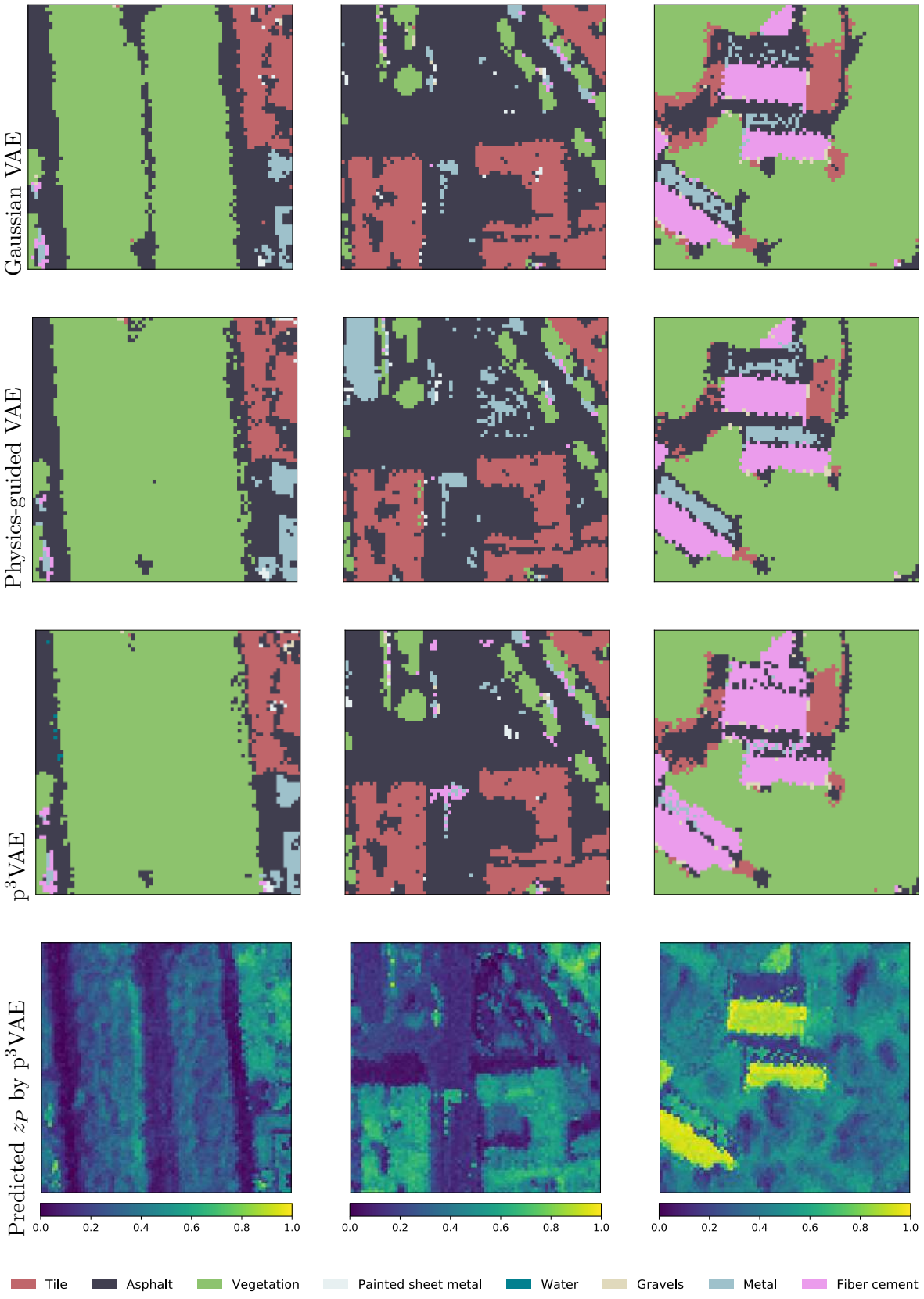


Figure 5.14: False color composition of subsets of the AI4GEO hyperspectral images, predicted land cover maps and normalized direct irradiance z_P (between 0 and 1) predicted by p^{3}VAE. Maps used for qualitative evaluation as only a very partial ground truth is available over those areas.

Table 5.4: p^{3}VAE average F1 score

Inference technique	no gs[1]	gs[1]
Simulated data set		
$q_{\phi}(y\|\boldsymbol{x})$	0.86	0.86
$\arg\max_y p_{\boldsymbol{\theta}}(y\|\boldsymbol{x})$	0.92	0.93
Real data set		
$q_{\phi}(y\|\boldsymbol{x})$	0.88	0.90
$\arg\max_y p_{\boldsymbol{\theta}}(y\|\boldsymbol{x})$	0.73	0.84

[a] **gs** refers to the gradient stopping technique described in section 3.1.

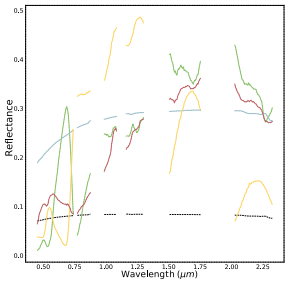


Figure 5.15: Estimated reflectance spectra of the class asphalt by p^{3}VAE without gradient stopping. Only spectrum is actually similar to a spectrum of asphalt.

3.4 Discussion

In this section, we discuss the performances of p^{3}VAE against other models, the particularities of its optimization and its relation to ϕ-VAE. Because our experiments showed that its interpolation capabilities are similar to conventional generative models, we focus on its extrapolation and disentanglement capacities which are greatly increased by its physical part.

3.4.1 Extrapolation capabilities

p^{3}VAE demonstrated high extrapolation capabilities in our experiments. On the simulated data set, great improvements over state-of-the-art models were made by computing predictions with $\arg\max_y p(y|\boldsymbol{x})$ (as described in section 3.1). As a matter of fact, the physics part f_P is explicitly used in this case. In contrast, the physics model is only used during optimization when we perform inference with the predictive distribution $q_{\phi}(y|\boldsymbol{x})$. On the contrary, higher extrapolation performances were observed on the real data set by computing predictions with $q_{\phi}(y|\boldsymbol{x})$. This is likely due to the higher *intrinsic* intra-class variability of real data, especially in a urban environment. As a matter of fact, materials are not lambertian in reality and most pixels are not pure (*e.g.* asphalt can be mixed with manhole covers or cars, north-facing roofs can be covered by moss, gravels can be mixed with sparse grass, and so on...). Therefore, there is a shift between the train and the test data distribution caused by *intrinsic* intra-class variability. As far as those variations of reflectance are not caused by variations in the irradiance conditions and are not captured in the training data set, p^{3}VAE sometimes fails to compute a faithful likelihood $p_{\boldsymbol{\theta}}(\boldsymbol{x}|y, \boldsymbol{z}_A, z_P)$, which causes more confusions when we compute predictions with $\arg\max_y p(y|\boldsymbol{x})$. Therefore, the less the assumptions under which the model is considered perfect are verified, the less accurate is the inference using $\arg\max_y p(y|\boldsymbol{x})$.

Experiments on the simulated data set have shown that the physics part of p^{3}VAE makes a great difference in handling data variations caused by variations in the irradiance conditions. The simulated test data set was generated with out-of-distribution physical factors, *i.e.* test irradiance conditions out of the range of the training irradiance conditions. In particular, the pixels of sheet metal (both labeled and unlabeled) were only seen under the same irradiance conditions during training. For this reason, the CNN and the conventional semi-supervised models failed to predict sheet metal pixels under roughly 20% of the test irradiance conditions using the predictive distribution. However, predictions from the gaussian VAE using $\arg\max_y p(y|\boldsymbol{x})$ approximately reduced by half the confusions over the sheet metal class. Be-

sides, the CNN with exhaustive annotations was still worse than gaussian VAE and p^{3}VAE on the simulated data. This illustrates that generative models as a whole can truly graspe some factors of variation that generalize to out-of-distribution data, in contrast to the predictive distribution only. In particular, the physics part of p^{3}VAE highly increased the extrapolation performances for classes with little inter-class similarities (*e.g.* only 3% of mistakes for the sheet metal class), since the physics model can extrapolate by nature. The limits of p^{3}VAE are reached when two spectra intrinsically share spectral features, such as tile and sandy loam: the model has similar or slightly smaller accuracy than a gaussian VAE, but higher and more meaningful uncertainty, as discussed below. In that sense, experiments on the real data set seem to show that a discriminative model, such as the CNN, can have a slightly higher discriminative power when a large labeled data set is available.

The physics-guided VAE made worse predictions with $\arg\max_y p(y|\boldsymbol{x})$ than with the predictive distribution on the simulated data. We provide an explanation through the analysis of the JEMMIG metric in the next subsection. Concerning ssInfoGAN, it learned to generate realistic samples but it failed to fully capture both the *intrinsic* and *physics* intra-class variabilities, as we can see on Fig. 5.13. In addition, realistic samples do not necessarily imply realistic relationships between samples and classes, despite the mutual information term between samples and latent codes in the ssInfoGAN objective.

We can derive an uncertainty estimate $\sigma(z_P)$ over $\delta_{dir}cos\,\Theta$ from p^{3}VAE, which yields easily explainable predictions. In other words, p^{3}VAE can tell us that different classes are likely, but under different conditions. Uncertainty estimates over latent variables grounded to physical properties are naturally more explainable than uncertainty estimates over abstract variables. From a user perspective, this is a valuable feature. Besides, the entropy of the predictive distribution $H(q_\phi(y|\boldsymbol{x}))$ gives a complementary uncertainty estimate. In fact, $H(q_\phi(y|\boldsymbol{x}))$ and $\sigma(z_P)$ poorly correlate ($R^2 \approx 0.2$ in our experiments), because the classifier is generally uncertain in the shadows although z_P is well inferred.

If p^{3}VAE successfully correlates spectral variations induced by variations of irradiance with z_P, one main limitation is that it cannot accurately estimate the true irradiance conditions. The inversion of the direct irradiance conditions and of the surface reflectance is indeed an ill-posed problem.

3.4.2 Disentanglement

We evaluated the disentanglement of the latent space qualitatively and quantitatively. A major difference that we should highlight is that the quantitative approach evaluates how a variation of the factors induces a variation of the latent code. The qualitative approach on the other hand evaluates how a change in the latent code induces a change in the observation. Therefore, assessing the disentanglement qualitatively requires expert knowledge.

Qualitatively, p^{3}VAE showed higher disentanglement capabilities than other generative models. The decoder architecture, with the machine learning part and the physical part, naturally induces a disentanglement of the latent space. Generated samples with varying z_P values automatically yields physically meaningful variations of spectra. Quantitatively, the JEMMIG metric highlights the rough approximation of the true diffuse irradiance as far as p^{3}VAE estimates the same Ωp_Θ given a $\delta_{dir}cos\Theta$ estimate: this is the reason why the mutual information gap is very low for the diffuse irradiance factor Ωp_Θ. In contrast, the mutual information gap for the direct irradiance factor is very high, which highlights the natural disentanglement induced by the physics model which grounds z_P to $\delta_{dir}cos\,\Theta$. We believe that the joint entropy term for the direct irradiance factor is large because the bias between the predicted

z_P and the true $\delta_{dir} cos\,\Theta$ depends on the class, which leads to an unclear relation between the factor and the code when the class y is not known. Concerning the physics-guided VAE and ssInfoGAN, they suffered from mode collapse during our experiments on the simulated data. Concerning the physics-guided VAE, the very low mutual information gap for $\delta_{dir} cos\,\Theta$ shows that the physics-guided VAE failed to capture this factor of variation. As a consequence, the model predicts approximately the same $\delta_{dir} cos\,\Theta$ for any spectrum. This is why it had mediocre performances using $\arg\max_y p_{\boldsymbol{\theta}}(y|\boldsymbol{x})$ compared to other generative models. This also explains why the physics-guided VAE had rather low joint entropies for the irradiance factors. Concerning ssInfoGAN, visual inspection of the generated samples (see Fig. 5.13) corroborates the low joint entropy: the model only learned one mode per class (*e.g.* it only generated healthy grass samples when conditioned on the *Vegetation* class and only one type of tile when conditioned on the *Tile* class). Thus, in the case of mode collapse (which causes a low joint entropy of factors and codes), the JEMMIG metric can be misleading and we argue that the mutual information gap is more relevant in this case.

Finally, it seems that there is a strong link between disentanglement and segmentation accuracy. Precisely, we measured a 0.44 correlation coefficient between the mutual information gap and the average F1 score on the simulated data. Intuitively though, this correlation would decrease with a decrease of the physics intra-class variability at the expense of an increase of the intrinsic and semantic intra-class variabilities.

3.4.3 Optimization

Loss landscape. In our experiments, we had more trouble to reach convergence with p^{3}VAE than other models. In particular, the training loss for some configurations of weights at initialization systematically diverged, irrespective to the learning rate. We guess that the physics model implicitly constrains the admissible solutions of the machine learning part, which leaves the optimization in a local minima from the beginning, for some specific weight initialization.

Gradient stopping. The ablation study showed that gradient stopping can have a significant impact on p^{3}VAE, either on the accuracy of the model, or on the relations it infers between the class labels and the estimated reflectance spectra. On the simulated data, similar performances observed with or without gradient stopping highlight that some values of (y, $\boldsymbol{z}_A$, z_P) can lead to good reconstructions even with very bad reflectance estimates. In this case though, p^{3}VAE looses physical sense and interpretability. On the real data set, however, a very large decrease of accuracy was observed (when predictions were made with $\arg\max_y p(y|\boldsymbol{x})$) without gradient stopping. We believe that this difference between simulated and real data can be explained by a larger noise and a higher *intrinsic* intra-class variability on real data: the decoder might be more prone to diverge with inconsistent values of $(\boldsymbol{z}_A, z_P, y)$. Overall, gradient stopping is crucial to regularize the machine learning part f_A of p^{3}VAE.

3.4.4 Independence of z_A and z_P

In the framework of p^{3}VAE, we assumed that 1) $p(\boldsymbol{z}_A, \boldsymbol{z}_P, y) = p(\boldsymbol{z}_A)p(\boldsymbol{z}_P)p(y)$ and that 2) $q_{\boldsymbol{\phi}}(\boldsymbol{z}_A, \boldsymbol{z}_P|\boldsymbol{x}, y) = q_{\boldsymbol{\phi}}(\boldsymbol{z}_A|\boldsymbol{x}, y)q_{\boldsymbol{\phi}}(\boldsymbol{z}_P|\boldsymbol{x}, y)$. However, those assumptions may not be true depending on the application. Concerning the assumption 1), it is indeed an unrealistic assumption for the land cover mapping of urban areas. For instance, pixels of tile are likely to have medium to high irradiance conditions (depending on the inclination of the roofs) while

pixels of asphalt are likely to have very low irradiance conditions (roads in the shadows) or high irradiance conditions (roads in the sun). However, having a more realistic prior on the latent variables does not seem trivial while we weakly weighted the KL divergence between the prior and the posterior distributions in our experiments, giving low importance to the prior. Concerning the assumption 2), there indeed may have correlations between $\boldsymbol{z}_A$ and $\boldsymbol{z}_P$ conditioned on $\boldsymbol{x}$ and y for some materials. For instance, roofs facing North are more likely to have a bit moss than roofs facing South, which induces intrinsic variations that are correlated with the irradiance conditions. We believe though that the dependence that may exist between $\boldsymbol{z}_A$ and $\boldsymbol{z}_P$ can be neglected in this case. In cases where the dependence would be strong, further connections between the framework of physics-integrated VAEs, which condition $\boldsymbol{z}_P$ on $\boldsymbol{z}_A$, and p^{3}VAE could be studied.

4 Conclusions & perspectives

■ **Conclusions** We introduced p^{3}VAE, a hybrid generative model that combines conventional neural networks with a phyics model. In this framework, the physical part partially explains the true underlying factors of data variation, which naturally leads to a disentangled latent space and to high extrapolation capabilities for illumination conditions that are not present in the training set. The gradient stopping technique to train p^{3}VAE in a semi-supervised setting led to a good use of the machine learning part, which was reflected with physically meaningful inferred latent variables and uncertainty estimates. The introduced inference technique empirically demonstrated its benefits on a simulated remote sensing scene, showing that high extrapolation performances are reached when the physical model is explicitly used during the inference, provided that the conditions under which the physical model is assumed perfect are verified. In any case, experiments on a real hyperspectral image demonstrated the large gains of accuracy p^{3}VAE brings against conventional semi-supervised generative models and supervised convolutional neural networks when few data is labeled and when a large part of the intra-class variability comes from different irradiance conditions. Therefore, in order to use p^{3}VAE for the land cover mapping of a metropolis, experiments with a greater number of classes with higher *intrinsic* and *semantic* intra-class variability should be done.

■ **Perspectives** In future work, we would like to enhance p^{3}VAE to handle spatial context, which we believe would help the inference of the irradiance conditions, in conjunction to tighter a priori regularization on z_P.

p^{3}VAE could also be used for other applications, such as the classification of the seabed into clear and shallow waters, by introducing the modeling of the optical path through the water, or for the characterization of methane plume.

The inversion of the methane concentration indeed falls in p^{3}VAE formalism: the observed radiance from the airborne sensor depends on the land cover and on the methane concentration, that is the acquisition conditions. The impact of the methane plume on the signal can be numerically modeled with radiative transfer models: methane absorbs the upward radiant flux for some wavelengths in the SWIR. Let's denote $\boldsymbol{x}$ the observed radiance, y the land cover class, $\boldsymbol{z}_A$ a latent variable that encodes spectral *intrinsic* and *semantic* intra-class variability, and z_P a latent variable that relates to the methane concentration ρ_{CH4}. Then, the likelihood of p^{3}VAE would be defined as follows:

$$p_{\boldsymbol{\theta}}(\boldsymbol{x}|y, z_P, \boldsymbol{z}_A) = \mathcal{N}\Big(\boldsymbol{x}|f_P(f_A(y, \boldsymbol{z}_A), z_P), \Sigma\Big) \tag{5.29}$$

where $f_A : (y, \boldsymbol{z}_A\ ;\ \boldsymbol{\theta}) \longmapsto \hat{\boldsymbol{\rho}}$ and $f_P = f_P^2 \circ f_P^1 : (\hat{\boldsymbol{\rho}}, z_P\ ;\ R_{atm}, I_{tot}, \tau_{atm}, S)$ where R_{atm} is the radiance scattered by the atmosphere, I_{tot} is the total irradiance, τ_{atm} is the upward

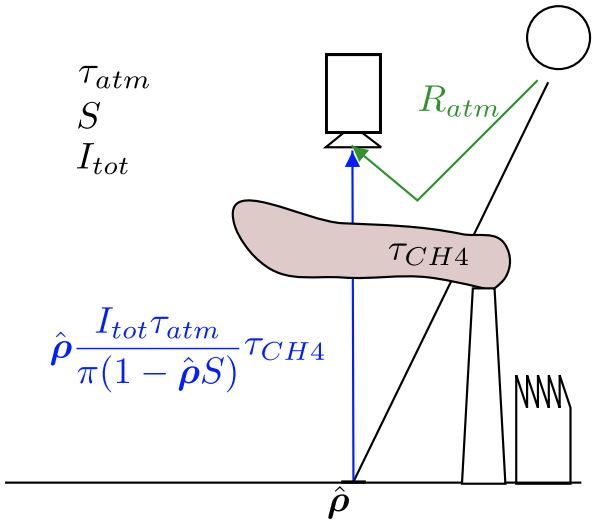


Figure 5.16: Illustration of the radiant flux through a methane plume

atmospheric transimttance and S is the atmospheric spherical albedo. The different terms are illustrated in Fig. 5.16. More precisely, f_P^1 is a numerical code such as COMANCHE [Poutier et al., 2002] that gives the transmittance through the methane plume and f_P^2 is an analytical model:

$$f_P^1(z_P\,;\,R_{atm},\,L_{atm},I_{tot},\tau_{atm},S) = \tau_{CH_4}(z_P) \tag{5.30}$$

$$f_P^2(\hat{\boldsymbol{\rho}},f_P^1(z_P)\,;\,R_{atm},I_{tot},\tau_{atm},S) = R_{atm} + \hat{\boldsymbol{\rho}}\frac{I_{tot}\tau_{atm}}{\pi(1-\hat{\boldsymbol{\rho}}S)}f_P^1(z_P) \tag{5.31}$$

The work presented in this chapter has resulted in a publication under revision in Springer Machine Learning:

- R. Thoreau, L. Risser, V. Achard, B. Berthelot and X. Briottet, "p^{3}VAE: a physics-integrated generative model. Application to the semantic segmentation of optical remote sensing images," arXiv preprint arxiv.org/abs/2210.10418

5 Appendices

5.1 Hybrid model likelihood

In section 3.2, we defined the likelihood of our model as follows:

$$p_{\boldsymbol{\theta}}(x|y, z_P, z_A) := \frac{1}{Z} f(x)\mathcal{S}(x) \tag{5.32}$$

where $f(x) = \mathcal{N}(x|f_P(f_A(y, z_A), z_P), \sigma^2 I)$, $\mathcal{S}(x) = exp(-\lambda\, arccos(\frac{x^T\mu}{\|x\|\|\mu\|}))$ with $\sigma, \lambda \in \mathbb{R}$ some hyperparameters and Z a finite constant such that the density integrates to one. Let us prove that such a constant $Z := \int f(x) \cdot \mathcal{S}(x)dx$ exists.

First, we can easily show that $f : [0,1]^B \longrightarrow \mathbb{R}$ is square-integrable, as well as $\mathcal{S} : [0,1]^B \longrightarrow \mathbb{R}$:

$$\int |\mathcal{S}(x)|^2 dx = \int_{[0,1]^B} exp(-2\lambda\, arccos(\frac{x^T\mu}{\|x\|\|\mu\|}))dx \tag{5.33}$$

$$\leq \int_{[0,1]^B} dx = 1 \tag{5.34}$$

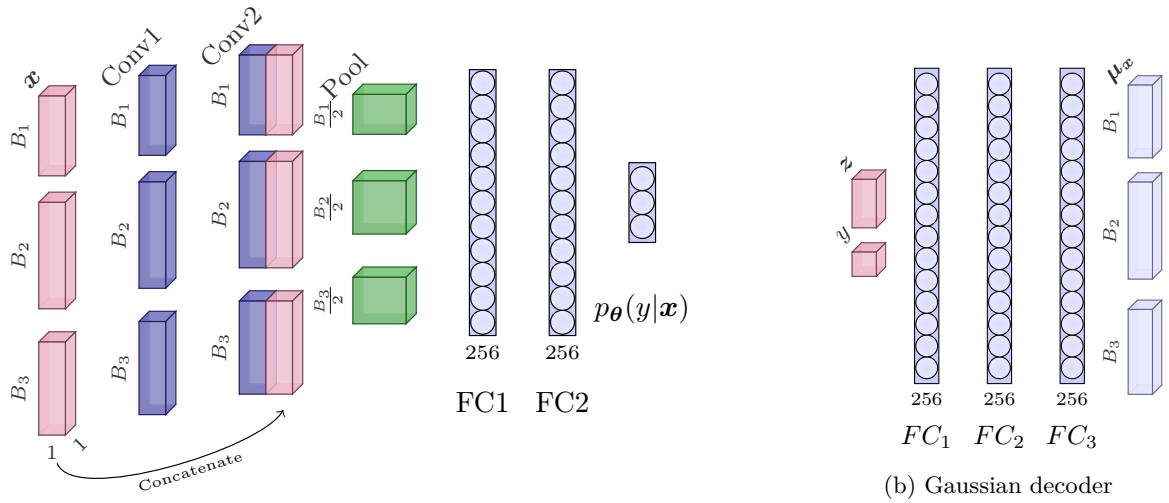
since $exp(-2\lambda\, arccos(\frac{x^T\mu}{\|x\|\|\mu\|})) \leq 1$ for all μ, $x \in [0,1]^B$. Moreover, f and $\mathcal{S}$ being continous over $[0,1]^B$, the Cauchy-Schwarz inequality implies that:

$$\left| \int f(x) \cdot \mathcal{S}(x)dx \right| \leq \left(\int f(x)^2 dx \right)^{\frac{1}{2}} \left(\int \mathcal{S}(x)^2 dx \right)^{\frac{1}{2}} \tag{5.35}$$

$$= C \in \mathbb{R} \tag{5.36}$$

Thus, $Z \in \mathbb{R}$ and $p_{\boldsymbol{\theta}}(x|y, z_P, z_A)$ properly defines a probability density function.

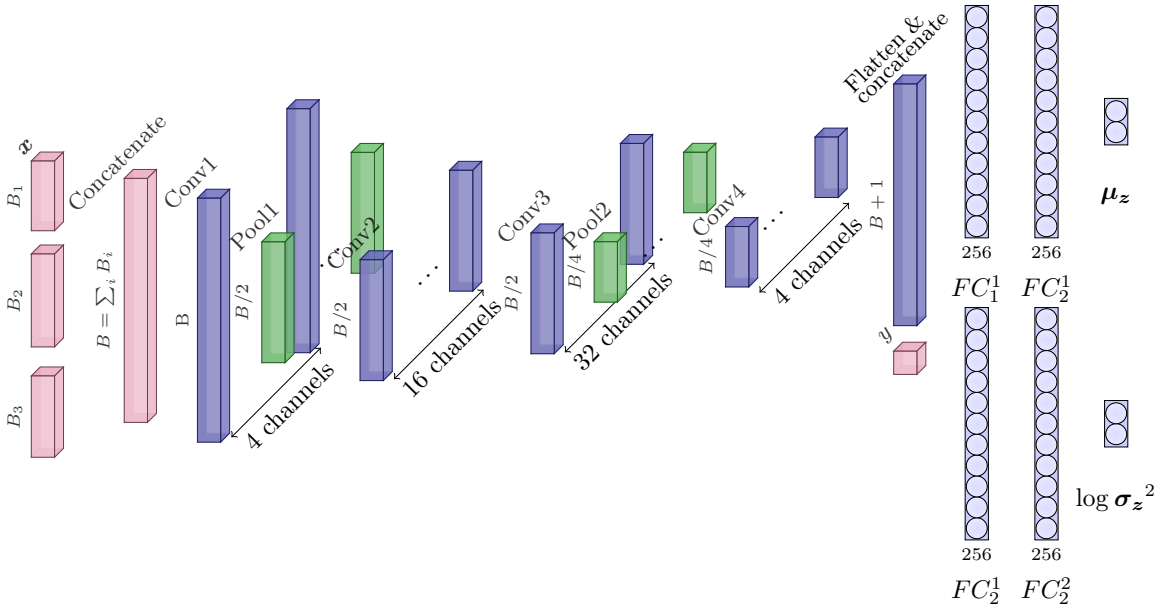
5.2 Model architectures



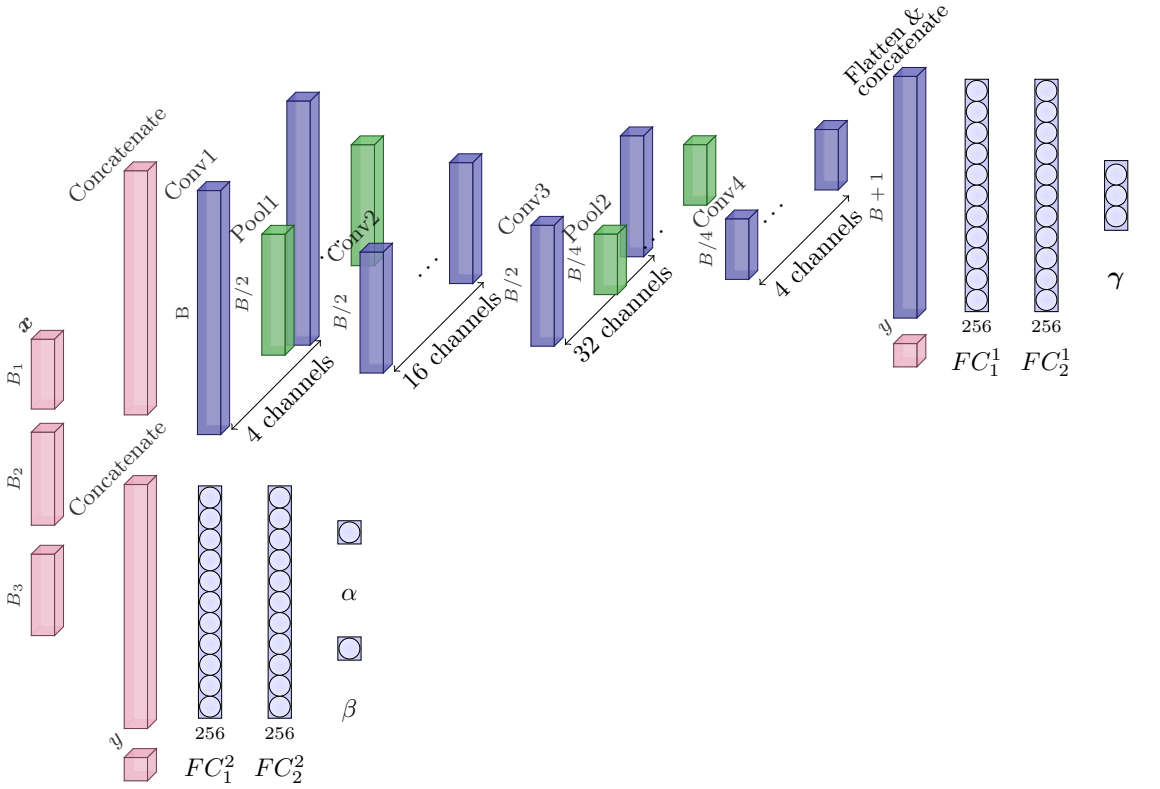
(a) CNN

(b) Gaussian decoder

Figure 5.17: Illustration of the models architecture

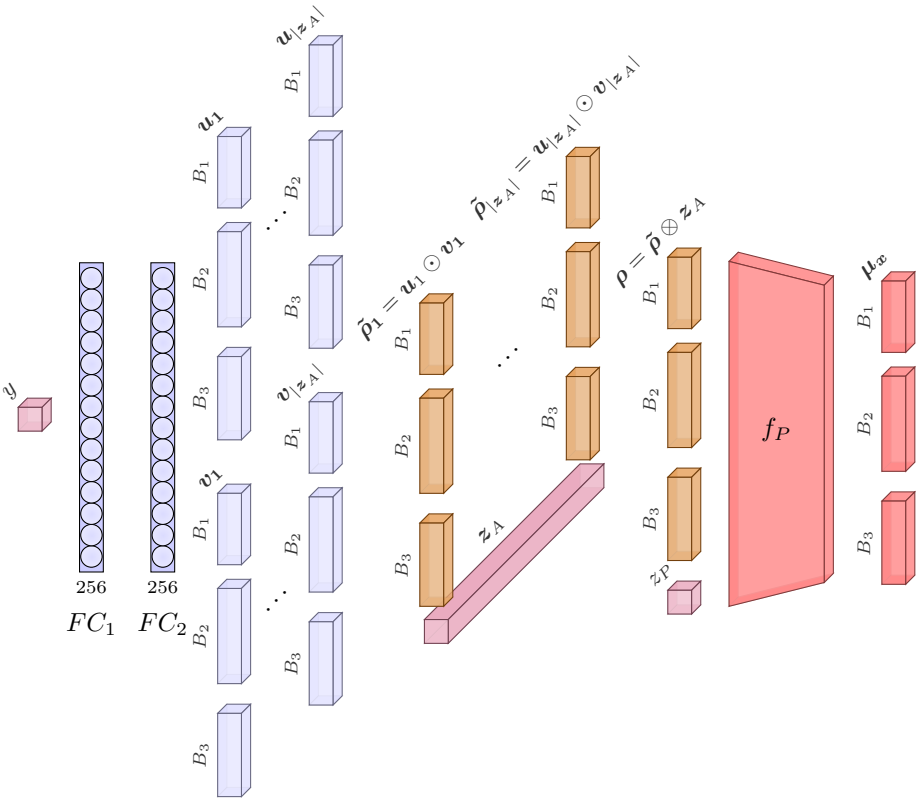


(c) Gaussian encoder

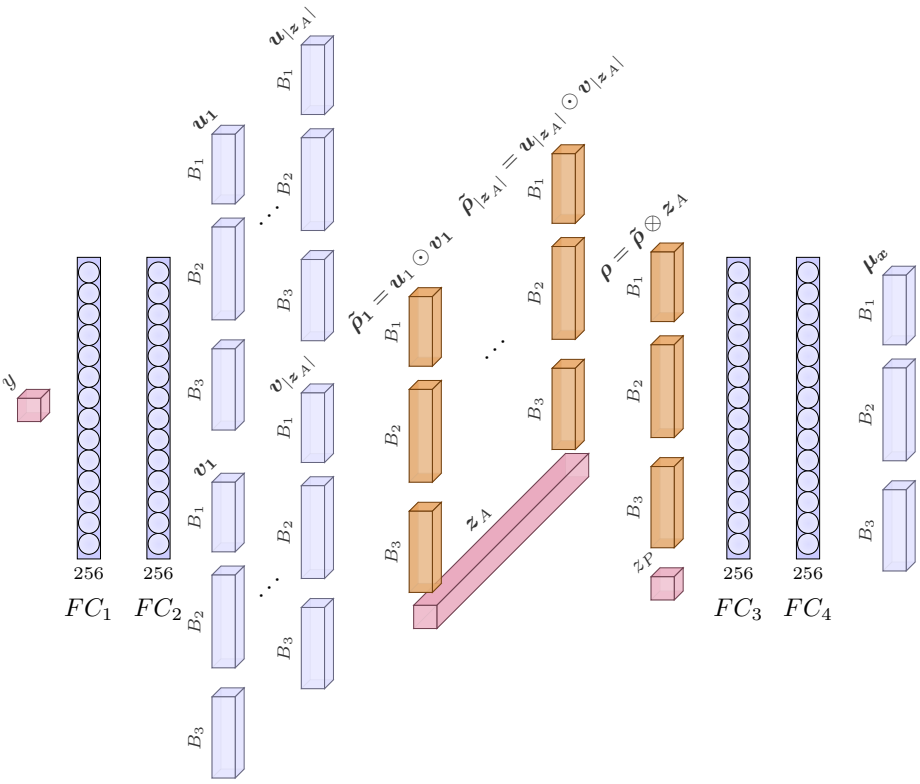


(d) Physics-guided encoder

Figure 5.17: Illustration of the models architecture

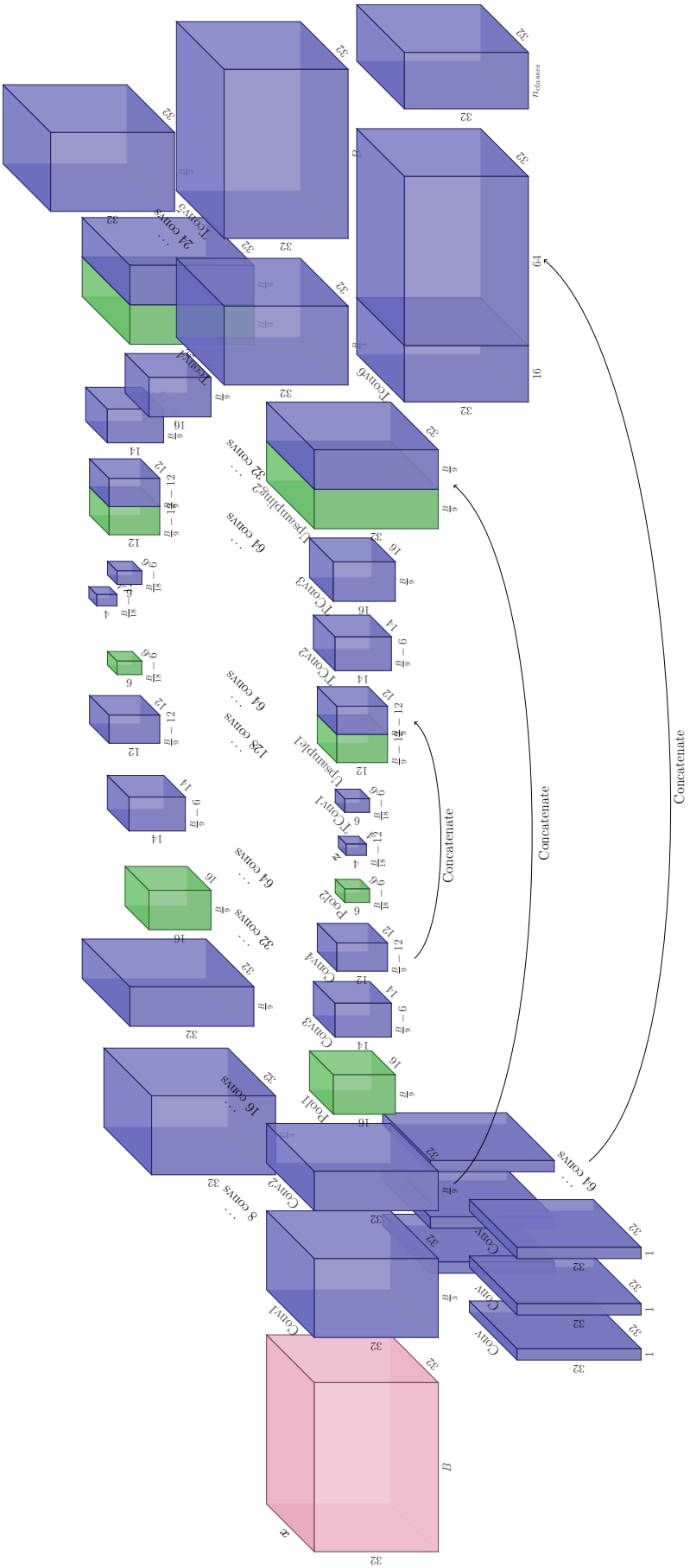


(e) Hybrid decoder



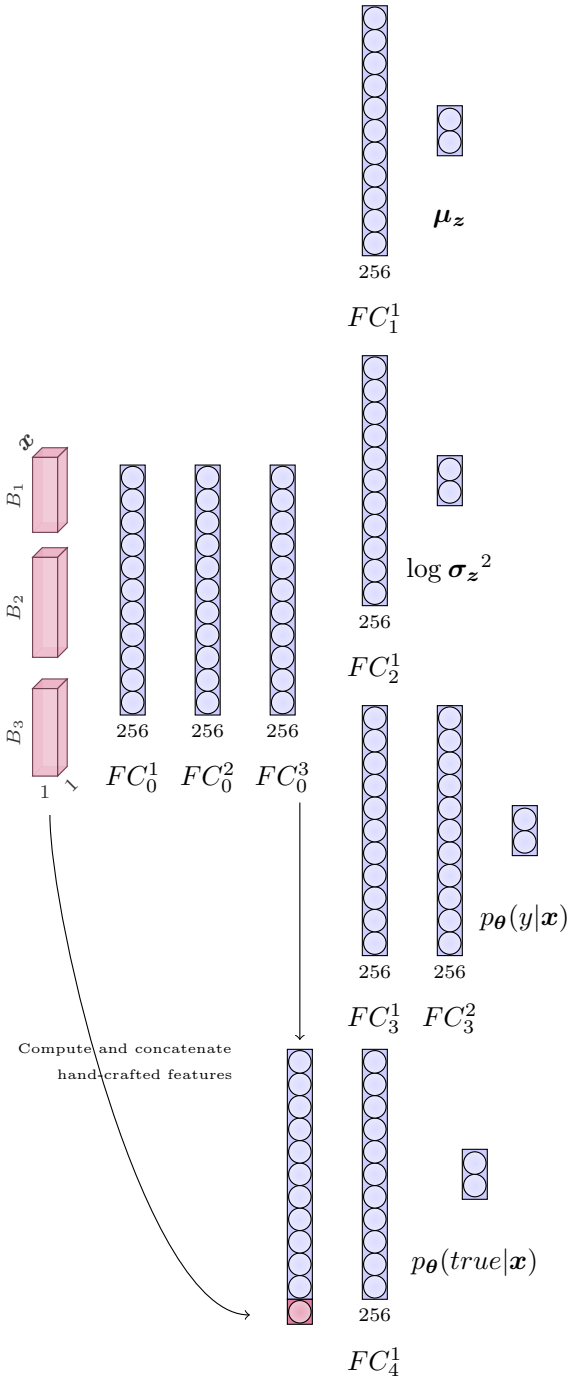
(f) Physics-guided decoder

Figure 5.17: Illustration of the models architecture

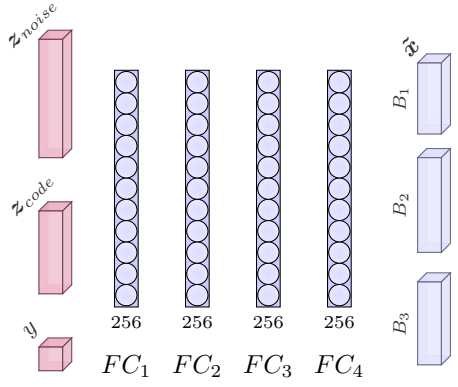


(g) FG-Unet

Figure 5.17: Illustration of the models architecture



(h) GAN encoder



(i) GAN generator

Figure 5.17: Illustration of the models architecture

6 References

Aragon-Calvo, M. A. and Carvajal, J. (2020). Self-supervised learning with physics-aware neural networks–i. galaxy model fitting. *Monthly Notices of the Royal Astronomical Society*, 498(3):3713–3719. 114, 122

Arjovsky, M., Chintala, S., and Bottou, L. (2017). Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR. 130

Audebert, N., Le Saux, B., and Lefèvre, S. (2019). Deep learning for classification of hyperspectral data: A comparative review. *IEEE geoscience and remote sensing magazine*, 7(2):159–173. 130

Castillo-Navarro, J., Le Saux, B., Boulch, A., Audebert, N., and Lefèvre, S. (2021). Semi-supervised semantic segmentation in earth observation: The minifrance suite, dataset analysis and multi-task network study. *Machine Learning*, pages 1–36. 131

Chen, C., Zheng, G., Wei, H., and Li, Z. (2020). Physics-informed generative adversarial networks for sequence generation with limited data. In *NeurIPS Workshop on Interpretable Inductive Biases and Physically Structured Learning*. 122

Chen, R. T. Q., Li, X., Grosse, R. B., and Duvenaud, D. K. (2018). Isolating sources of disentanglement in variational autoencoders. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc. 123

Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., and Abbeel, P. (2016). Infogan: Interpretable representation learning by information maximizing generative adversarial nets. *Advances in neural information processing systems*, 29. 123, 130

Cheung, B., Livezey, J. A., Bansal, A. K., and Olshausen, B. A. (2015). Discovering hidden factors of variation in deep networks. In Bengio, Y. and LeCun, Y., editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Workshop Track Proceedings*. 123

Gal, Y. and Ghahramani, Z. (2016). Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In Balcan, M. F. and Weinberger, K. Q., editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1050–1059, New York, New York, USA. PMLR. 130

Gastellu-Etchegorry, J.-P., Grau, E., and Lauret, N. (2012). Dart: A 3d model for remote sensing images and radiative budget of earth surfaces. *Modeling and simulation in engineering*, (2). 125

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems*, 27. 123

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. C. (2017). Improved training of wasserstein gans. *Advances in neural information processing systems*, 30. 130

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., and Lerchner, A. (2017). beta-vae: Learning basic visual concepts with a constrained variational framework. In *International conference on learning representations*. 123

Hu, W., Huang, Y., Wei, L., Zhang, F., and Li, H. (2015). Deep convolutional neural networks for hyperspectral image classification. *Journal of Sensors*, 2015:1–12. 130

Karaletsos, T., Belongie, S., and Rätsch, G. (2015). Bayesian representation learning with oracle constraints. *arXiv preprint arXiv:1506.05011*. 123

Kim, H. and Mnih, A. (2018). Disentangling by factorising. In *International Conference on Machine Learning*, pages 2649–2658. PMLR. 123

Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization. In Bengio, Y. and LeCun, Y., editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. 130

Kingma, D. P., Mohamed, S., Jimenez Rezende, D., and Welling, M. (2014). Semi-supervised learning with deep generative models. *Advances in neural information processing systems*, 27. 120, 121, 127, 130

Kingma, D. P. and Welling, M. (2014). Auto-encoding variational bayes. In Bengio, Y. and LeCun, Y., editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*. 120

Kulkarni, T. D., Whitney, W. F., Kohli, P., and Tenenbaum, J. (2015). Deep convolutional inverse graphics network. *Advances in neural information processing systems*, 28. 123

Kumar, A., Sattigeri, P., and Balakrishnan, A. (2017). Variational inference of disentangled latent concepts from unlabeled observations. *arXiv preprint arXiv:1711.00848*. 123

Linial, O., Ravid, N., Eytan, D., and Shalit, U. (2021). Generative ode modeling with known unknowns. In *Proceedings of the Conference on Health, Inference, and Learning*, pages 79–94. 122

Liu, Z., Luo, P., Wang, X., and Tang, X. (2015). Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*. 123

Locatello, F., Bauer, S., Lucic, M., Raetsch, G., Gelly, S., Schölkopf, B., and Bachem, O. (2019). Challenging common assumptions in the unsupervised learning of disentangled representations. In *international conference on machine learning*, pages 4114–4124. PMLR. 123

Miesch, C., Poutier, L., Achard, V., Briottet, X., Lenot, X., and Boucher, Y. (2005). Direct and inverse radiative transfer solutions for visible and near-infrared hyperspectral imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 43(7):1552–1562. 115, 127

Mitchell, T. M. (1980). *The need for biases in learning generalizations*. 114

Poutier, L., Miesch, C., Lenot, X., Achard, V., and Boucher, Y. (2002). Comanche and cochise: two reciprocal atmospheric codes for hyperspectral remote sensing. In *2002 AVIRIS Earth Science and Applications Workshop Proceedings*, pages 1059–0889. Jet Propulsion Laboratory Pasadena, CA, USA. 144

Raissi, M., Perdikaris, P., and Karniadakis, G. E. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378:686–707. 114, 122

Rezaabad, A. L. and Vishwanath, S. (2020). Learning representations by maximizing mutual information in variational autoencoders. In *2020 IEEE International Symposium on Information Theory (ISIT)*, pages 2729–2734. IEEE. 123

Rodriguez, E. G. (2021). On disentanglement and mutual information in semi-supervised variational auto-encoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1257–1262. 123

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer. 130

Roupioz, L., Briottet, X., Adeline, K., Al Bitar, A., Barbon-Dubosc, D., Barda-Chatain, R., Barillot, P., Bridier, S., Carroll, E., Cassante, C., Cerbelaud, A., Déliot, P., Doublet, P., Dupouy, P., Gadal, S., Guernouti, S., De Guilhem De Lataillade, A., Lemonsu, A., Llorens, R., Luhahe, R., Michel, A., Moussous, A., Musy, M., Nerry, F., Poutier, L., Rodler, A., Riviere, N., Riviere, T., Roujean, J., Roy, A., Schilling, A., Skokovic, D., and Sobrino, J. (2023). Multi-source datasets acquired over toulouse (france) in 2021 for urban microclimate studies during the camcatt/ai4geo field campaign. *Data in Brief*, 48:109109. 127

Roussel, G., Weber, C., Briottet, X., and Ceamanos, X. (2017). Comparison of two atmospheric correction methods for the classification of spaceborne urban hyperspectral data depending on the spatial resolution. *International Journal of Remote Sensing*, 39(5):1593–1614. 116

Spurr, A., Aksan, E., and Hilliges, O. (2017). Guiding infogan with semi-supervision. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 119–134. Springer. 127, 130

Stoian, A., Poulain, V., Inglada, J., Poughon, V., and Derksen, D. (2019). Land cover maps production with high resolution satellite image time series and convolutional neural networks: Adaptations and limits for operational systems. *Remote Sensing*, 11(17):1986. 127, 130, 131

Subramaniam, A., Wong, M. L., Borker, R. D., Nimmagadda, S., and Lele, S. K. (2020). Turbulence enrichment using physics-informed generative adversarial networks. *arXiv preprint arXiv:2003.01907*. 122

Takeishi, N. and Kalousis, A. (2021). Physics-integrated variational autoencoders for robust and interpretable generative modeling. *Advances in Neural Information Processing Systems*, 34:14809–14821. 114, 119, 122, 123

von Rueden, L., Mayer, S., Beckh, K., Georgiev, B., Giesselbach, S., Heese, R., Kirsch, B., Walczak, M., Pfrommer, J., Pick, A., et al. (2021). Informed machine learning-a taxonomy and survey of integrating prior knowledge into learning systems. *IEEE Transactions on Knowledge & Data Engineering*, (01):1–1. 114

Wang, S., Teng, Y., and Perdikaris, P. (2021). Understanding and mitigating gradient flow pathologies in physics-informed neural networks. *SIAM Journal on Scientific Computing*, 43(5):A3055–A3081. 122

Wei, C., Zhang, J., and Wu, C. (2020). Thermodynamic consistent neural networks for learning material interfacial mechanics. In *NeurIP workshop*. 122

Whitney, W. F., Chang, M., Kulkarni, T., and Tenenbaum, J. B. (2016). Understanding visual concepts with continuation learning. *arXiv preprint arXiv:1602.06822*. 123

Yildiz, C., Heinonen, M., and Lahdesmaki, H. (2019). Ode2vae: Deep generative second order odes with bayesian neural networks. *Advances in Neural Information Processing Systems*, 32. 122

Zhao, S., Ren, H., Yuan, A., Song, J., Goodman, N., and Ermon, S. (2018). Bias and generalization in deep generative models: An empirical study. *Advances in Neural Information Processing Systems*, 31. 114

Zheng, Q., Zeng, L., and Karniadakis, G. E. (2020). Physics-informed semantic inpainting: Application to geostatistical modeling. *Journal of Computational Physics*, 419:109676. 122

Chapter 6

Self-supervised learning: improving spectral representations from unlabeled data

Contents

1 Chapter summary

In chapter 4, we have shown that AL techniques can guide the annotation of hyperspectral images towards informative training data sets for semantic segmentation. However, given a limited number of pixels to label, AL algorithms are bound to bring only partial information on the intra-class variability. Therefore, we have introduced p^{3}VAE in chapter 5, a hybrid model that jointly leverages a priori physics knowledge with unlabeled data to capture *physics* intra-class variability with limited labeled samples. Nevertheless, the benefits of p^{3}VAE may be lessened by the prevalence of *intrinsic* intra-class variability over *physics* intra-class variability (in the following, we will refer to *intrinsic* variability as the combination of *intrinsic* and *semantic* variabilities). In this context, leveraging the unlabeled data to cope with the *intrinsic* intra-class variability without human interaction seems crucial. A legitimate question though, is whether spectral information from unlabeled data can be useful for semantic segmentation, insofar as *intrinsic* spectral variations are, as their name implies, intrinsic to the materials. It is therefore fundamental to find a relevant unsupervised task that will benefit to the segmentation task. Lately, self-supervision has raised considerable attention in the machine learning community as well as in the hyperspectral community. Self-supervised techniques aim to learn generic data representations by minimizing a pretext supervised loss, for which automatic labels can be generated, that can then be fine-tuned for downstream tasks.

Problematic

To what extent can self-supervision exploit unlabeled data to learn class-specific representations without class supervision?

■ Summary of contributions

⋆ **Introduction and analysis of a large hyperspectral data set**. Current public hyperspectral data sets (such as those used in former chapters) are not totally appropriate to address key issues in self-supervised and semi-supervised learning for hyperspectral image segmentation because they are not provided with appropriate train / test splits for semi-supervision. Therefore, we present in section 2 a large data set built from an airborne hyperspectral image of Toulouse, France, that stands out from other public data sets in the following aspects:

- A very high spatial resolution (1 m) and spectral resolution ($\leq$ 12 nm) from 0.4 µm to 2.5 µm,
- A large hierarchical nomenclature containing 32 land cover classes,
- A land use nomenclature containing 12 classes,
- 8 spatially disjoint splits of the ground truth in a labeled training set, an unlabeled training set, a validation set and a test set,
- Sparse annotations gathering approximately $380,000$ labeled pixels over a 90 km^2 area.

Thanks to its high spatial resolution, wide spatial coverage, large spectral domain, large nomenclature and large number of annotated samples, we believe that this data set well represents the spectral diversity and intra-class variability of land cover materials in a metropolis, which is supported by a qualitative comparison of our data set with other public data sets in section 2.2.

⋆ **A preliminary study on self-supervised tasks**. Many self-supervised tasks rely on data augmentation, which in essence, aims to produce new samples of the same example with different contexts. While spatial data augmentation techniques have been applied on hyperspectral images, spectral data augmentation are likely not faithful to intra-class spectral variations. Besides, intra-class spectral variations induce changes of low abstraction in contrast to spatial contextual variations that induce high level changes. With this in mind and on the basis that the combination of spectral features (for instance, an absorption peak in the SWIR combined to a steep increase of reflectance in the visible domain) are highly informative of the surface material, we adapt in section 3.1 the Masked Autoencoder (MAE) self-supervision strategy [He et al., 2022] to hyperspectral data. The aim of MAE is to learn representations from very small proportions of the attributes of data samples such that the decoder can reconstruct the whole input data. We empirically study its discriminative potential combined with non-parametric classification algorithms on the Toulouse data set in section 3.3.

⋆ **Introduction of ensembles of Prototypical Networks**. In conjunction with self-supervision techniques, it appears that class prototypes (*i.e.* as many spectra, or rather low-dimensional vectors, as there are classes, that can be used to perform non-parametric classification based on their distance with samples) could be appropriate to partially supervise the training of MAEs. In particular, the non-parametric few-shot classification scheme of Prototypical Networks [Snell et al., 2017] has interesting properties for conventional classification tasks, such as robustness to class imbalance. Few-shot techniques aim to learn machine learning models that can adapt to new classes from few labeled samples only. For a conventional classification task, we describe in section 3.2 how Prototypical Networks can be used to easily build ensembles of classifiers for inference, which is to our knowledge, not considered in the literature. We study their performance against conventional machine learning models on the Toulouse data set in section 3.3, though the experiments on the combination of Prototypical Networks with MAEs are not yet complete.

2 Toulouse Hyperspectral Data Set

2.1 Annotation of a large land cover ground truth

2.1.1 Data annotation methodology

As mentioned in chapter 3, reflectance spectra of diverse materials in the city of Toulouse were acquired with ASD spectrometers in the range of 0.4 µm to 2.5 µm, during the CAMCATT-AIGEO field campaign [Roupioz et al., 2023]. In-situ measures of the reflectance spectra of *clear paving stone*, *brown paving stone* and *red porous concrete* with pictures of the materials are shown in Fig. 6.1 as examples. These in-situ measurements have served as a basis to define a land cover nomenclature and to build a ground truth by photo-interpretation, additional field campaigns as well as with the help of exogenous data. Precisely, we used the "Registre Parcellaire Graphique"[1], a geographical information system that informs the crop type of agricultural plots over France, to annotate cultivated fields.

In total, we define the land cover nomenclature with 32 classes, dividing into 16 impermeable materials and 16 permeable materials, that we organize in a hierarchical nomenclature as shown in Fig. 6.2. 382,114 pixels are labeled with a land cover class. In contrast to con-

[1] https://artificialisation.developpement-durable.gouv.fr/bases-donnees/registre-parcellaire-graphique

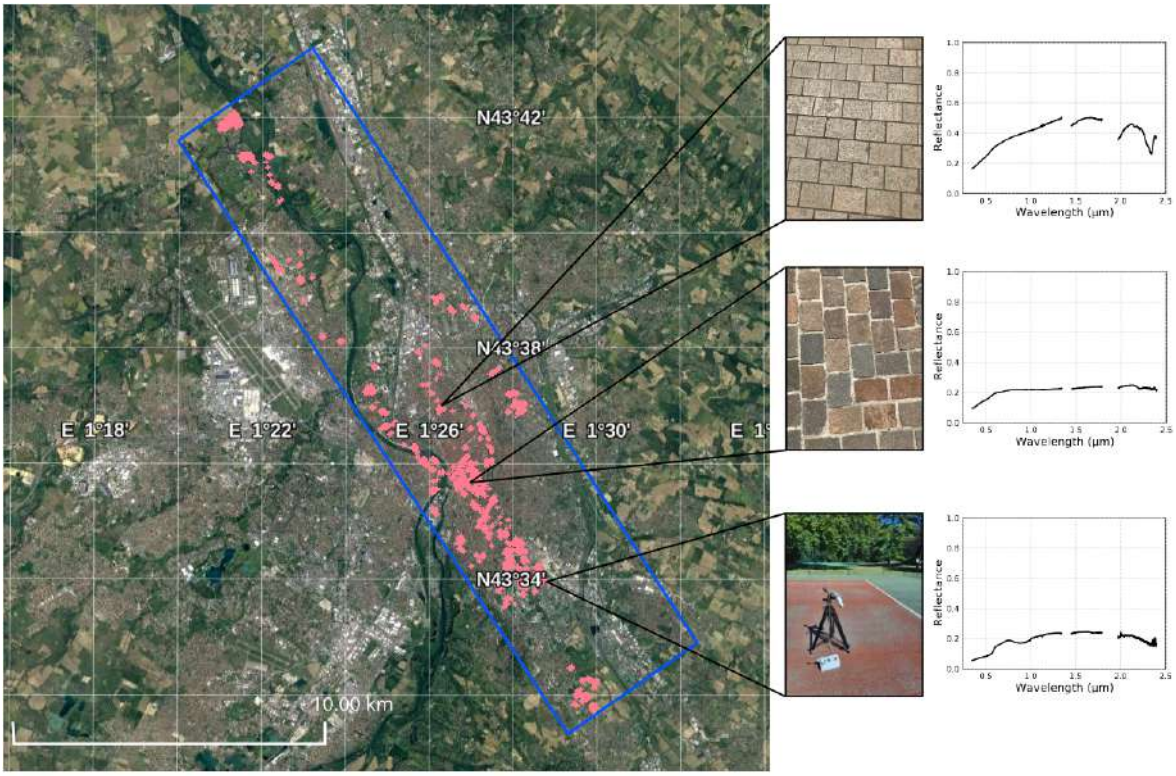


Figure 6.1: Area of Toulouse covered by the AI4GEO airborne hyperspectral image (in blue), our annotated ground truth (in red), and examples of reflectance spectra (clear paving stone, brown paving stone and red porous concrete, from top to bottom) measured on field with ASD spectrometers during the CAMCATT-AIGEO field campaign [Roupioz et al., 2023].

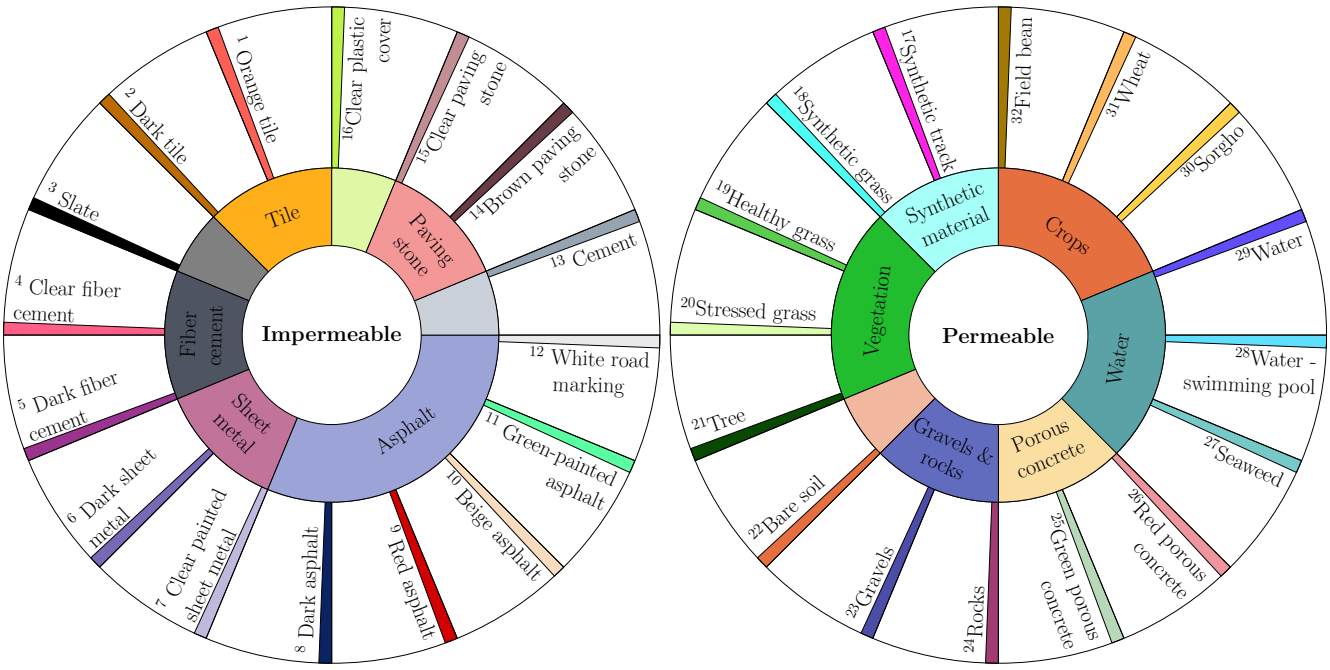
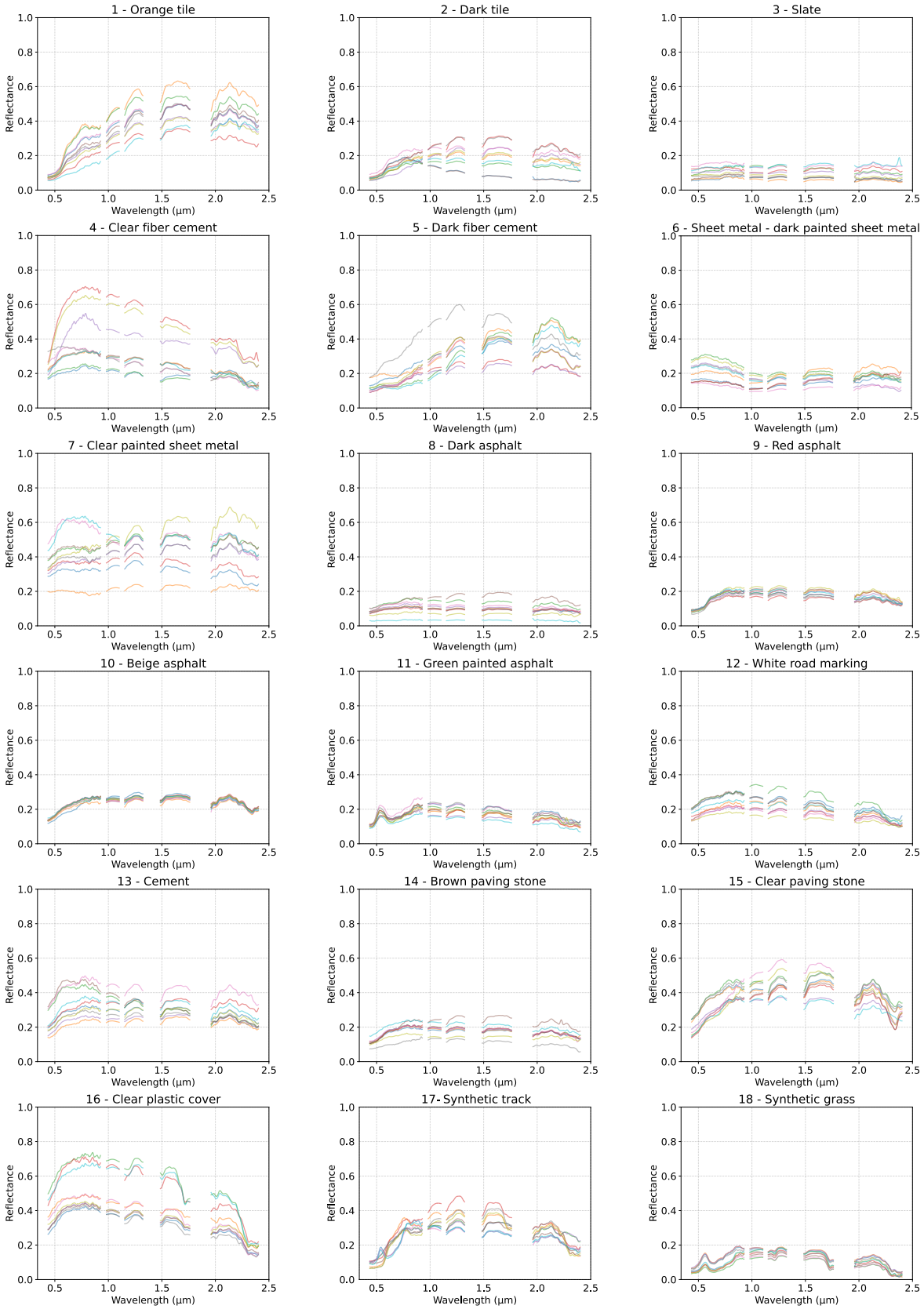


Figure 6.2: Land cover nomenclature of Toulouse Hyperspectral Data Set

ventional semantic segmentation data sets, our ground truth is made of sparse annotations, *i.e.* polygons that are disconnected from each other as shown in Fig. 6.4. We annotated the pixels with particular attention to the exactness of the land cover labels. In particular, we omitted from the ground truth, as much as possible, mixed pixels whose reflectance spectra are distorted by the presence of other materials within the pixels. Random subsets of spectra are plotted for each class in Fig. 6.3.

In addition to the land cover, we define a land use nomenclature which gathers more abstract semantic classes, listed in Tab. 6.1. Besides, we provide the direct and diffuse irradiance at ground level, as well as the solar zenith angle which is of 22.12°.


1 - Orange tile
2 - Dark tile
3 - Slate
4 - Clear fiber cement
5 - Dark fiber cement
6 - Sheet metal - dark painted sheet metal
7 - Clear painted sheet metal
8 - Dark asphalt
9 - Red asphalt
10 - Beige asphalt
11 - Green painted asphalt
12 - White road marking
13 - Cement
14 - Brown paving stone
15 - Clear paving stone
16 - Clear plastic cover
17- Synthetic track
18 - Synthetic grass
Reflectance
Wavelength (µm)

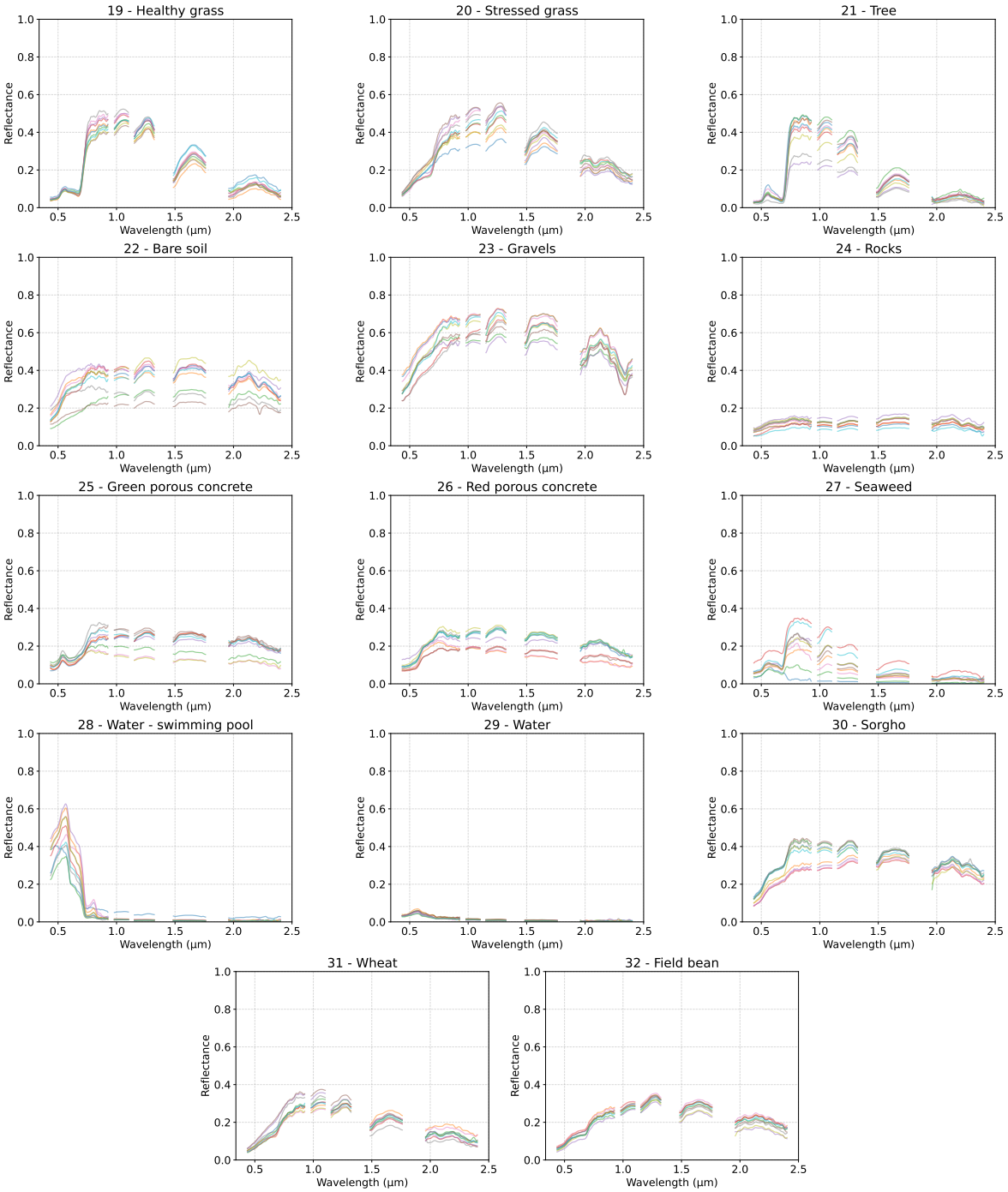


Figure 6.3: Random spectra of the Toulouse Hyperspectral Data Set

Figure 6.4: False-color composition and land cover ground truth over the Grand Rond

Table 6.1: Land use classes of Toulouse Hyperspectral Data Set.

#1	Roads	#7	Lakes / rivers/ harbors
#2	Railways	#8	Swimming pools
#3	Roofs	#9	Forests
#4	Parking lots	#10	Cultivated fields
#5	Building sites	#11	Boats
#6	Sport facilities	#12	Open areas

2.1.2 Ground truth split

Publicly available hyperspectral data sets have fueled a great deal of research. Nevertheless, if the Standardized Remote Sensing Data Website[2] of the IEEE Geoscience and Remote Sensing Society (GRSS) provides a set of community data sets and a tool to evaluate classifiers on undisclosed test samples, the ground truth of public data sets should be provided with standard training sets (divided in a subset for the supervised part and another subset for the self-supervised part) spatially disjoint to test sets, in order to properly evaluate and fairly compare self-supervised techniques. We emphasize that several works including [Audebert et al., 2019; Geiß et al., 2017; Lange et al., 2018] showed that random sampling of the training and test sets over-estimates the generalization performances of classifiers, which is partly explained by the fact that pixels belonging to the same semantic class but sampled in different geographical areas are obviously more likely to have different spectral signatures than neighboring pixels.

We provide 8 spatially disjoint splits such that the train and test sets are as independent as possible. Moreover, we split the training set in a labeled set and an unlabeled set to allow easy evaluations of semi-supervised techniques. The unlabeled set is itself split in a set that is guaranteed to exclusively contain pixels belonging to known classes, that we will denote as the labeled pool, and a set that contains truly unlabeled pixels and does not necessarily contain pixels belonging to classes in the nomenclature, that we will denote as the unlabeled pool. In order to guarantee that the train (labeled) set, the unlabeled set, the validation set and the test set do not overlap, we group neighboring polygons together in n_{groups} groups and define the ground truth split as a mixed integer problem:

$$\min_{u} \sum_{s\in\mathcal{S}} \sum_{i=1}^{n_{groups}} \sum_{k=1}^{c} P[i,k]\cdot u_{is} \tag{6.1}$$

$$\text{subject to: } \sum_{j=1}^{4} u_{ij} = 1 \quad \textit{i.e. each group should be at least in one set} \tag{6.2}$$

$$\forall s\in\mathcal{S}, \forall k\in\{1,\dots,c\}, \sum_{i=1}^{n_{groups}} P[i,k]\cdot u_{is} \geq p_s \cdot \sum_{i=1}^{n_{groups}} P[i,k] \tag{6.3}$$

$$\textit{i.e. for each class } k\textit{, the proportion of pixels in set } s \textit{ should be greater than the proportion } p_s$$

In this formulation, u_{ij} is 1 if group i is in set j, $P[i,k]$ is the number of pixels of class k in group i and $\mathcal{S}=\{1,3,4\}$ where the indices 1, 2, 3 and 4 correspond to the labeled training set, the labeled pool, the validation set and the test set, respectively (while the unlabeled pool is left out). In the provided splits, the proportion of samples within each set may vary depending on the class, and also differs given the split. In average, 13%, 29%, 14% and 46% of the labeled samples are in the labeled training set, the labeled pool, the validation set and the test set, respectively. In addition, the unlabeled pool contains nearly 2.6 million pixels. Hence, the labeled pixels used for training only represent 7% of all data.

The decision to divide the ground truth into splits of 13%, 29%, 14% and 46% of the ground truth stems from the following considerations, in order of priority: having a representative test set, having a representative validation set, having a sufficient number of samples in the training set for supervision to be relevant. However, the precise choice of the average proportions in each set is arbitrary and does not rely on a statistical analysis. In addition,

[2] http://dase.grss-ieee.org/index.php

we chose to provide only height splits of the ground truth because we could not find other solutions of the mixed integer problem that were significantly different from each other.

2.2 Data set analysis and comparison with other public hyperspectral data sets

We discuss several characteristics of our data set compared to currently public data sets.

Spectral and spatial variability In order to qualitatively compare the spectral and spatial complexity of hyperspectral data sets, we introduce a representation technique that applies on hyperspectral images with roughly the same spatial resolution but that may have different spectral range and spectral resolution. The representation technique is illustrated in Fig. 6.5 and works as follows. A collection of 64×64-pixel hyperspectral patches that are labeled, at least partially, are extracted from the image. Spectral indices, precisely the NDVI, ANVI, CI, NDVI_RE, VgNIR_BI, SAVI, are computed for each pixel of the patches (we select spectral indices that could be computed for every data sets, that is spectral indices that only use spectral channels included in the spectral domain of the Pavia image). 20 spectral bands are also uniformly sampled. Spatial features are computed with 24 predefined Gabor filters on the spectral average of the patch (4 different frequencies (from 1 m^{-1} to 10 m^{-1}) and 6 different orientations). Then, the spectral and spatial features are concatenated. Finally, patch-wise statistics are computed: the average, the standard deviation, the first and last deciles, the first and last quartiles, as well as the minimum and the maximum, yielding a 500-dimensional feature for each patch.

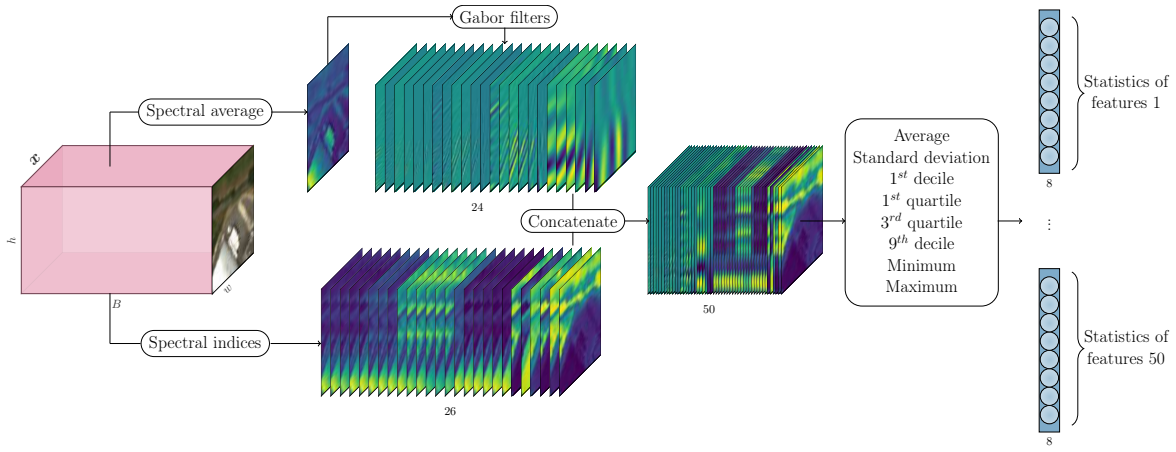


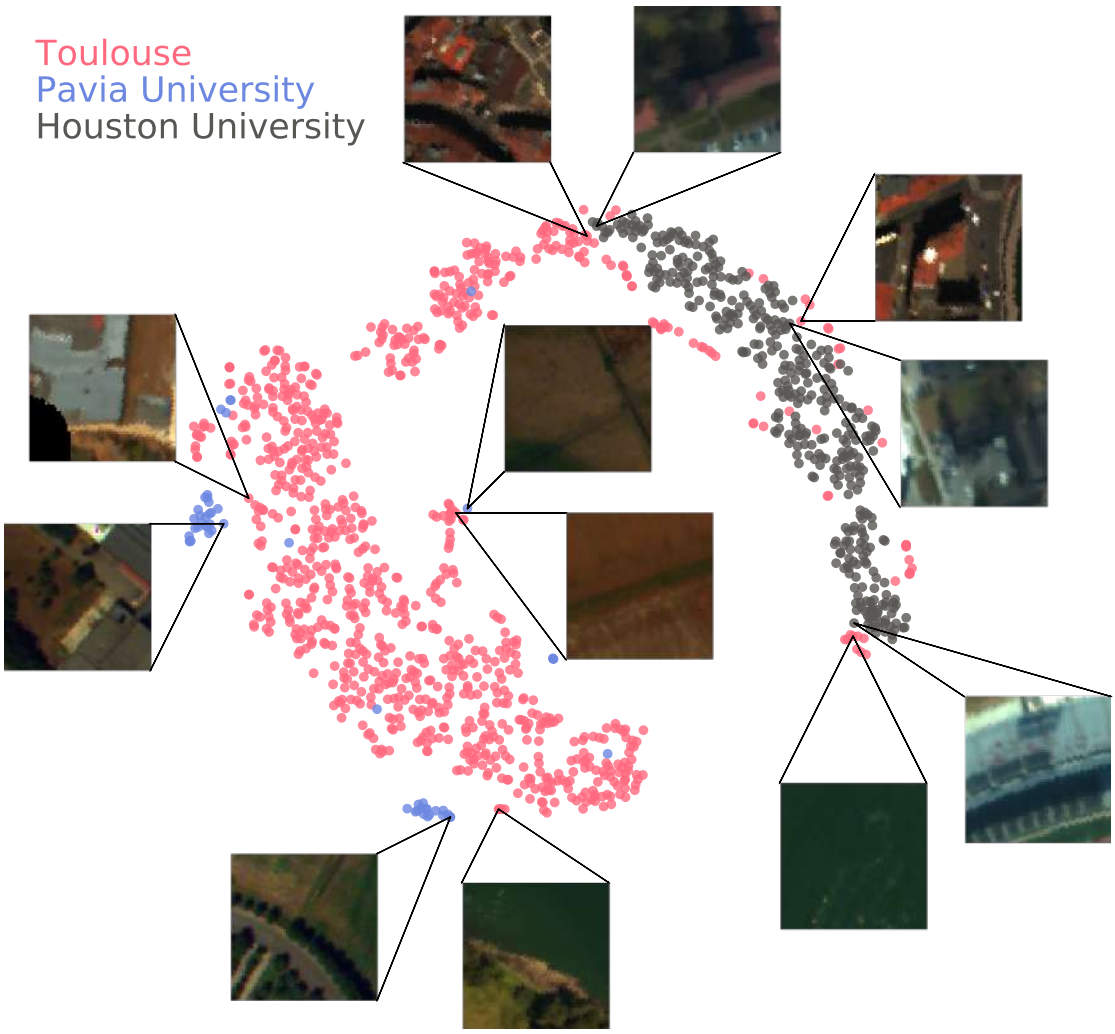
Figure 6.5: Illustration of our hand-crafted patch-wise feature extraction technique. The input is a 64 by 64 pixel hyperspectral patch. On one side, spectral indices (which include a selection of 20 spectral bands uniformly sampled along the spectral domain) are computed, resulting in 26 maps of 64 by 64 pixels. On the other side, the patch, averaged along the spectral dimension, is filtered by Gabor filters with 4 different frequencies (from 1 m^{-1} to 10 m^{-1}) and 6 different orientations, resulting in 24 maps. From every maps, spatial statistics are computed, resulting in a 500-dimensional feature.

We project in a 2 dimensional space the patch representations of Pavia university, Houston university and Toulouse data sets through a t-SNE [Van der Maaten and Hinton, 2008] transformation. The results are shown in Fig. 6.6: the Toulouse data set occupies more space than the others. Visualizing patches from different data sets that are close to each other in the 2-dimensional space shows that our hand-crafted representation method intuitively makes sense as far as most of those patches have comparable landscapes. We also separately visualize the 2D projections of spatial and spectral features, showing that most of the diversity of the

Toulouse data set comes from the spectral information. Besides, we additionally compute spectral features from the visible and near infrared only (as the image of Pavia university does not go beyond). From the 2D projection in Fig. 6.6c, we observe that the variability of Toulouse does not come from the additional spectral information of the SWIR domain, but seems to be the consequence of a larger variability of the landscapes.

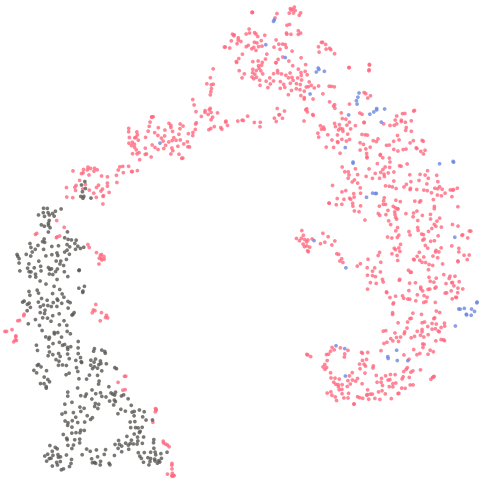
Long-tailed class imbalance Data sets with a long-tailed class distribution are data sets where a small number of classes account for a large part of samples while a large number of classes have only few examples [Zhang et al., 2023]. The difference between usual class imbalance and long-tailed class imbalance mainly lies in the number of classes with few samples. Fig. 6.7 shows that the Toulouse data set particularly exhibits this property, which is representative of life-like scenarios, compared to other hyperspectral data sets. In addition, we recall some statistics in Tab. 6.3 including the imbalance ratio, which is the ratio of the number of samples in the largest class over the number of samples in the smallest class.

Noisy labels Compared to the Houston data set, we argue that the Toulouse data set contains less noise in the ground truth. Although the Houston data set contains more than a half million labeled pixels, a substantial number of pixels are wrongly labeled, or are at least misleading as there are a mix of several materials. This noise in the ground truth is detrimental to classification models that put more emphasize on the spectral information rather than the spatial information. We show in Fig. 6.8 a few examples of noisy labels.



(a) Spatial + spectral (visible to SWIR) features

(b) Spatial features



(c) Spectral (visible to NIR) features

Figure 6.6: t-SNE projections of hand-crafted representations of 64×64 pixel hyperspectral patches from the Pavia University, Houston University and Toulouse data sets. (a) corresponds to the 2D projection of spatial-spectral features as described in Fig. 6.5, (b) corresponds to the 2D projection of the spatial features only, *i.e.* only the Gabor filters were used to represent the data and then the t-SNE projection was performed, and (c) corresponds to the 2D projection of the spectral features only, *i.e.* only the spectral indices were used to represent the data and then the t-SNE projection was performed. Moreover, only the smallest spectral domain, *i.e.* the spectral domain of the Pavia University image which covers the 0.4 µm - 0.86 µm range, was used in (c).

Table 6.2: Spectral and spatial characteristics of several hyperspectral data sets, including Toulouse.

Data set	**GSD**	**Spectral domain**	**Spectral resolution**
Indian Pines	20	0.4 µm - 2.5 µm	5 nm
Pavia University	1.3	0.43 µm - 0.86 µm	4 nm
Houston University	1	0.38 µm - 1.0 µm	3.5 nm
Toulouse	1	0.4 µm - 2.5 µm	3.5 nm (VNIR) & 12 nm (SWIR)

Table 6.3: Statistics of several hyperspectral data sets, including Toulouse.

Data set	**# classes**	**# data samples**	**Imbalance ratio**
Indian Pines	16	10249	123
Pavia University	9	42776	20
Houston University	20	512648	329
Toulouse	35	382, 114	225

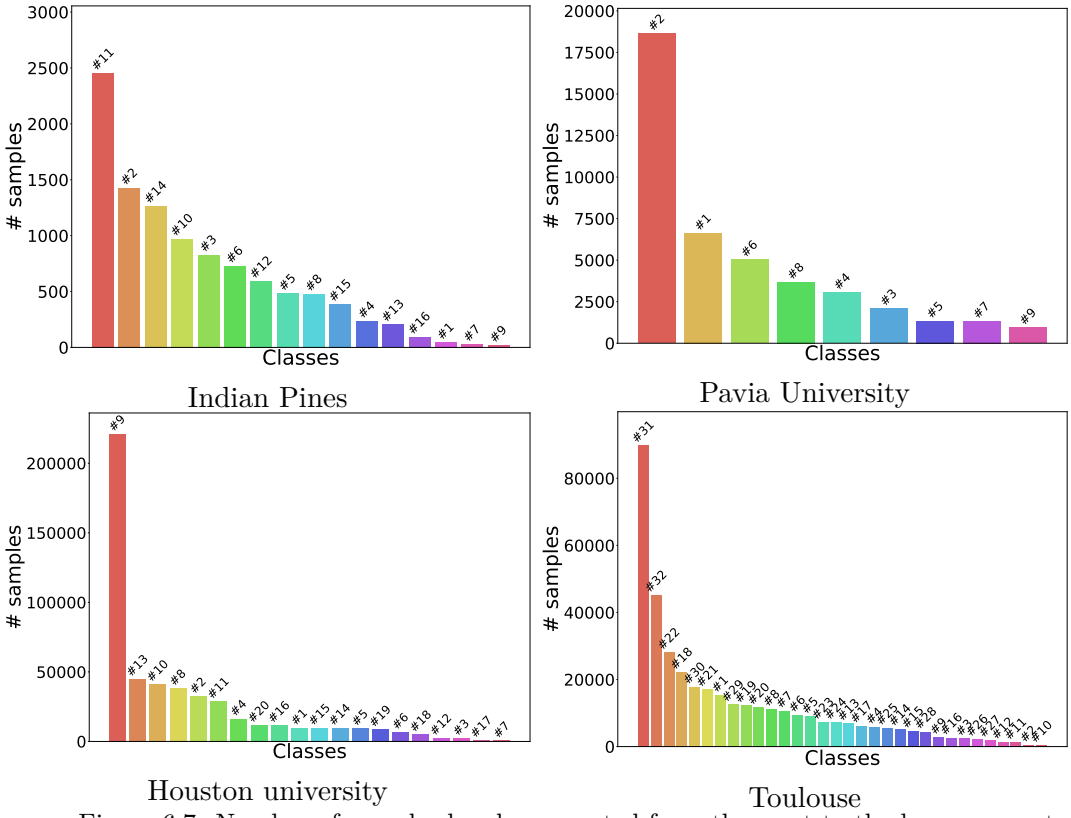


Figure 6.7: Number of samples by classes sorted from the most to the less represented.

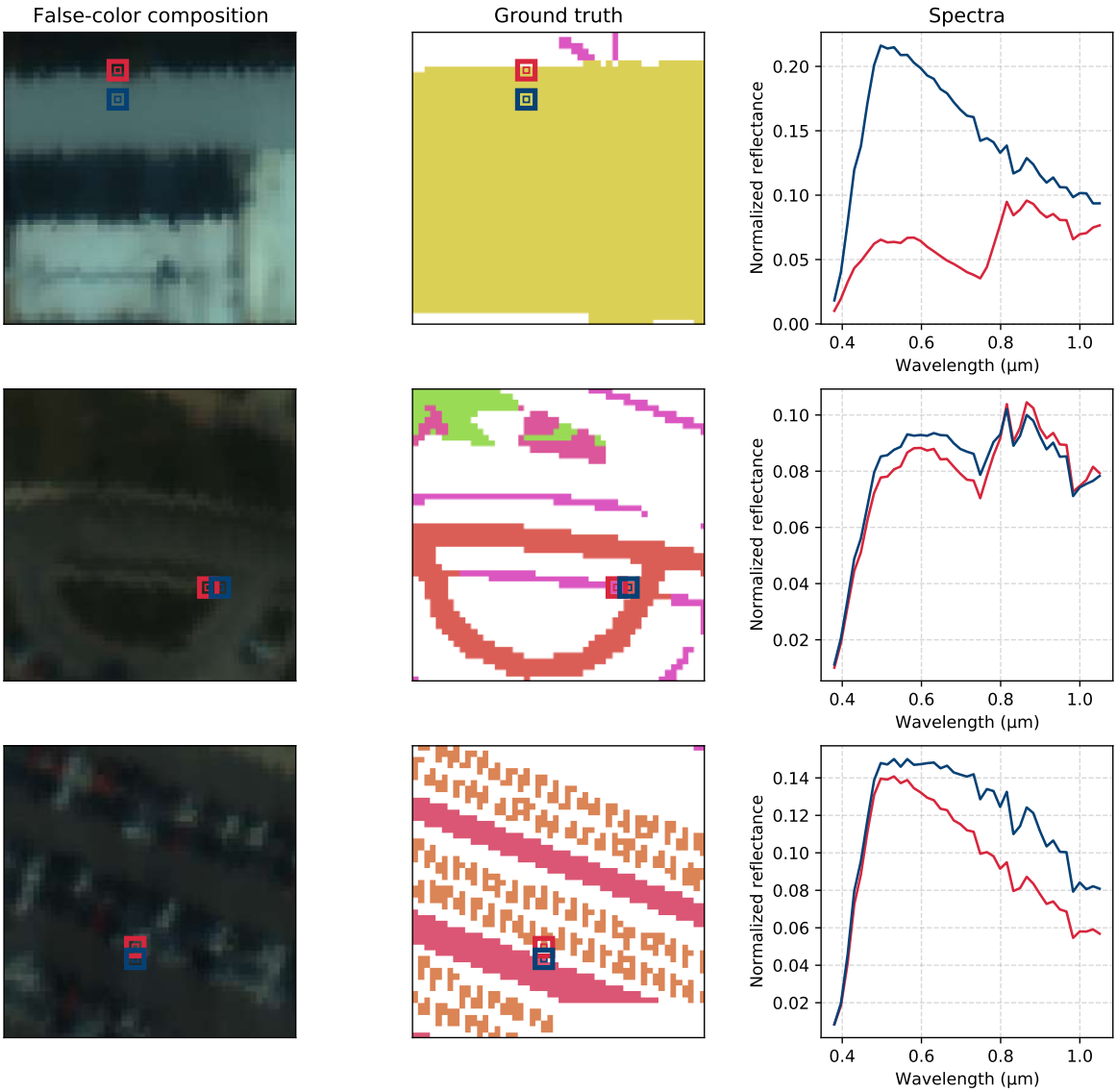


Figure 6.8: Illustration of noisy labels in the Houston university data set. The first row shows an example of two spectra labeled as *Non-residential buildings*. The pixel at the edge of the roof is mixed with the ground vegetation, which provides erroneous spectral information. The second and third rows show neighboring pixels labeled with different classes though they are actually a mixed of both classes.

Figure 6.9: Examples of annotations in the Toulouse data set. We paid attention to ignore mixed pixels such as pixels at edges of roofs or pixels mixed with small objects such as pipes on roofs.

3 Improving spectral representation learning with self-supervision and partial supervision from prototypes

In section 3.1, we review state-of-the-art self-supervised tasks and adapt the Masked Autoencoder [He et al., 2022] to hyperspectral data. Then, we introduce in section 3.2 an ensemble method based on prototypical networks to partially supervise training with labels. Finally, we carry numerical experiments on the Toulouse data set in section 3.3.

3.1 Finding a relevant self-supervised task to leverage unlabeled data

As we have seen in Chapter 2, common semi-supervised learning techniques jointly optimize machine learning models on a supervised task and on an auxiliary unsupervised task. A common choice for the auxiliary task is the reconstruction of the (high dimensional) data from a (low dimensional) representation. However, a wide range of new approaches, known as self-supervised learning techniques, have recently emerged in the machine learning community, by introducing more useful auxiliary tasks. Self-supervision consists in training the model on a supervised pretext task for which labels are automatically generated. In computer vision, pretext tasks include rotation self-supervision [Gidaris et al., 2018], exemplar self-supervision [Dosovitskiy et al., 2014], contrastive learning [Chen et al., 2020; Henaff, 2020; Oord et al., 2018; Tian et al., 2020; Wu et al., 2018], self-supervised knowledge distillation [Caron et al., 2021] or cluster-based self-supervision [Caron et al., 2018, 2020]. Rotation self-supervision aims to predict the rotation (0°, 90°, 180° or 270°) applied to an image. Exemplar self-supervision gathers transformations of the same data sample under one class and trains the model on the subsequent classification task. Contrastive learning consists in learning similar representations of automatically-generated pairs of data samples with common semantic properties (positive pairs), while learning dissimilar representations of unrelated data samples (negative pairs). In particular, the seminal framework of [Chen et al., 2020] is based on stochastic data augmentation (specifically the combination of random cropping, random color distortions and random Gaussian blur) and a contrastive loss function that aims to identify a positive pair within a batch of samples. Self-supervised knowledge distillation takes inspiration from knowledge distillation [Hinton et al., 2015] by taking a teacher network to supervise a student network that sees different transformations of the input data. Finally, cluster-based self-supervision consist in supervising a machine learning model with pseudo-labels derived from a clustering algorithm. Self-supervised learning has also been integrated in semi-supervised learning frameworks, such as [Zhai et al., 2019] or [Fini et al., 2023] that combine cluster-based self-supervision with class prototype learning.

Many self-supervised learning techniques have been directly applied to hyperspectral data, such as [Zhao et al., 2022] that augment hyperspectral patches with random cropping and random color distortions, and [Duan et al., 2022] that apply random rotations as well as spectral random noise and spectral mirroring, both in the framework of self-supervised contrastive learning. Here we shall note that the physical soundness of random color distortions and spectral mirroring should be questioned. Other works have introduced data augmentation techniques that are specific to hyperspectral data, such as [Qian et al., 2022] that creates positive pairs of data samples by sampling monochromatic images from neighboring spectral channels, or [Qin et al., 2023] that pairs spectrally close samples.

All in all, most attention has been put on learning spatial-spectral representations with self-supervised techniques based on data augmentation. While those techniques have experimentally demonstrated benefits in term of robustness to various spatial contexts (*e.g.* pose, orientation, background...) for natural images, few tasks seem appropriate to handle spectral

variations. Actually, we believe that finding a relevant spectral data augmentation technique to represent *physics* and *intrinsic* intra-class variations is not trivial. Even more, we argue that spectra cannot be augmented such that the induced variations would be faithful to *intrinsic* intra-class variabilities, as far as they are, as the name implies, intrinsic to the chemical composition of matter, and not dependent of a context.

Therefore, we focus on Masked Autoencoders (MAE) [He et al., 2022], a self-supervised technique that do not rely on data augmentation. The idea of MAE is to strongly mask the input data and to learn to reconstruct its missing parts. For reflectance spectra, as much as the combination of spectral features (absorption peaks, spectral inflexion, etc.) at different wavelengths is closely related to the chemical composition of the land surface, the *masked reconstruction* task of MAE seems particularly relevant to learn discriminative features without class supervision. While MAE were adapted to 21×21-pixel hyperspectral patches in [Zhu et al., 2023], we adapt the architecture of MAE to the spectral dimension only, illustrated in Fig. 6.10, that we will refer as SpectralMAE.

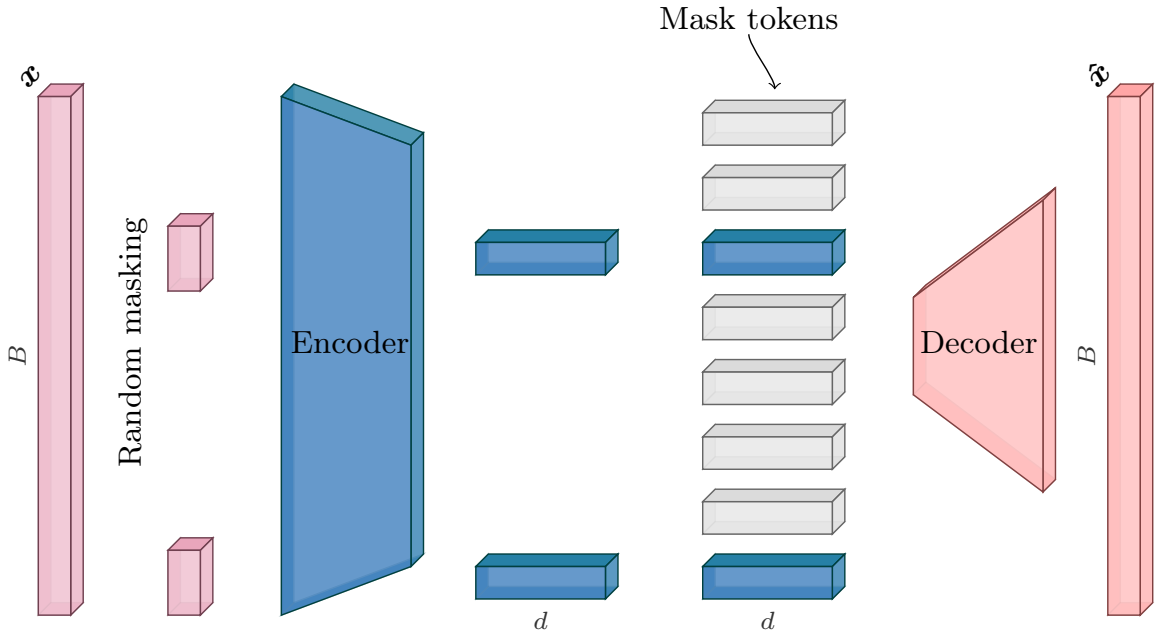


Figure 6.10: Illustration of the SpectralMAE adapted from [He et al., 2022]. The input spectrum is divided in small sequences. A large part of the sequences are randomly masked. The visible spectral sequences (with positional encoding) are encoded by the transformer. Then, learnable mask tokens are concatenated with the embeddings of the sequences, that are mapped to the reconstructed data by the light-weight decoder.

Originally, MAE processes RGB images that are divided in small patches which are encoded and decoded by vision transformers [Dosovitskiy et al., 2021]. Transformers [Vaswani et al., 2017] are neural networks, first introduced to process language, that integrate the relative position of words (or image patches) in order to learn correlations between features that are arbitrarily distant in sentences (or in images). While transformers have been adapted for hyperspectral data in [Hong et al., 2021], we use a simpler architecture and make as few changes as possible from the original MAE, keeping in mind two important points: 1) in contrast to words that are very abstract concepts (or to small image patches that can contain high-level information), reflectance values are not meaningful by themselves, 2) in contrast to words or to image patches, the relative distance between spectral channels does not contain semantic information. By this we mean that the position of a verb with regard to a noun in a sentence can be crucial for its meaning, while the distance between two spectral

features simultaneously observed on a spectrum is not informative. This is why we believe that the transformer architecture is not particularly relevant for hyperspectral data, but is very convenient for the masked reconstruction task.

In [He et al., 2022], 75% of the images are masked, leading to a non-trivial reconstruction task (as simply interpolating between image patches lead to very mediocre reconstructions). As far as the spectral dimension of our data is much lower than the spatial dimension of theirs, we opt for a stronger masking of 90%, that will be discussed in section 3.3.

3.2 Guiding representation learning with prototypes

In this section, we present Prototypical Networks [Snell et al., 2017] that were introduced for the task of few-shot learning as we think that they can guide self-supervised representation learning and have interesting properties for inference on a conventional classification task.

3.2.1 Prototypical Networks

Prototypical networks were introduced by [Snell et al., 2017] to perform few-shot learning. The task of few-shot classification consists in the classification of data which include samples that belong to unseen classes during training, with the help of only few additional (typically 1 to 10) labeled examples from those classes. A prototypical network $\pi_{\boldsymbol{\theta}}$ maps labeled data $\{\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(N)}\}$ to representations $\{\boldsymbol{z}^{(1)}, \dots, \boldsymbol{z}^{(N)}\}$ from which a subset is used to compute class prototypes. The prototype $\boldsymbol{p}_i$ of class i is the mean of the features $\pi_{\boldsymbol{\theta}}(\boldsymbol{x}^{(k)}) \in \mathcal{S}_i$, where $\mathcal{S}_i$ denotes a subset of labeled samples from class i, called the support set:

$$\boldsymbol{p}_i = \frac{1}{|\mathcal{S}_i|} \sum_{\boldsymbol{x}^{(k)} \in \mathcal{S}_i} \pi_{\boldsymbol{\theta}}(\boldsymbol{x}^{(k)}) \tag{6.4}$$

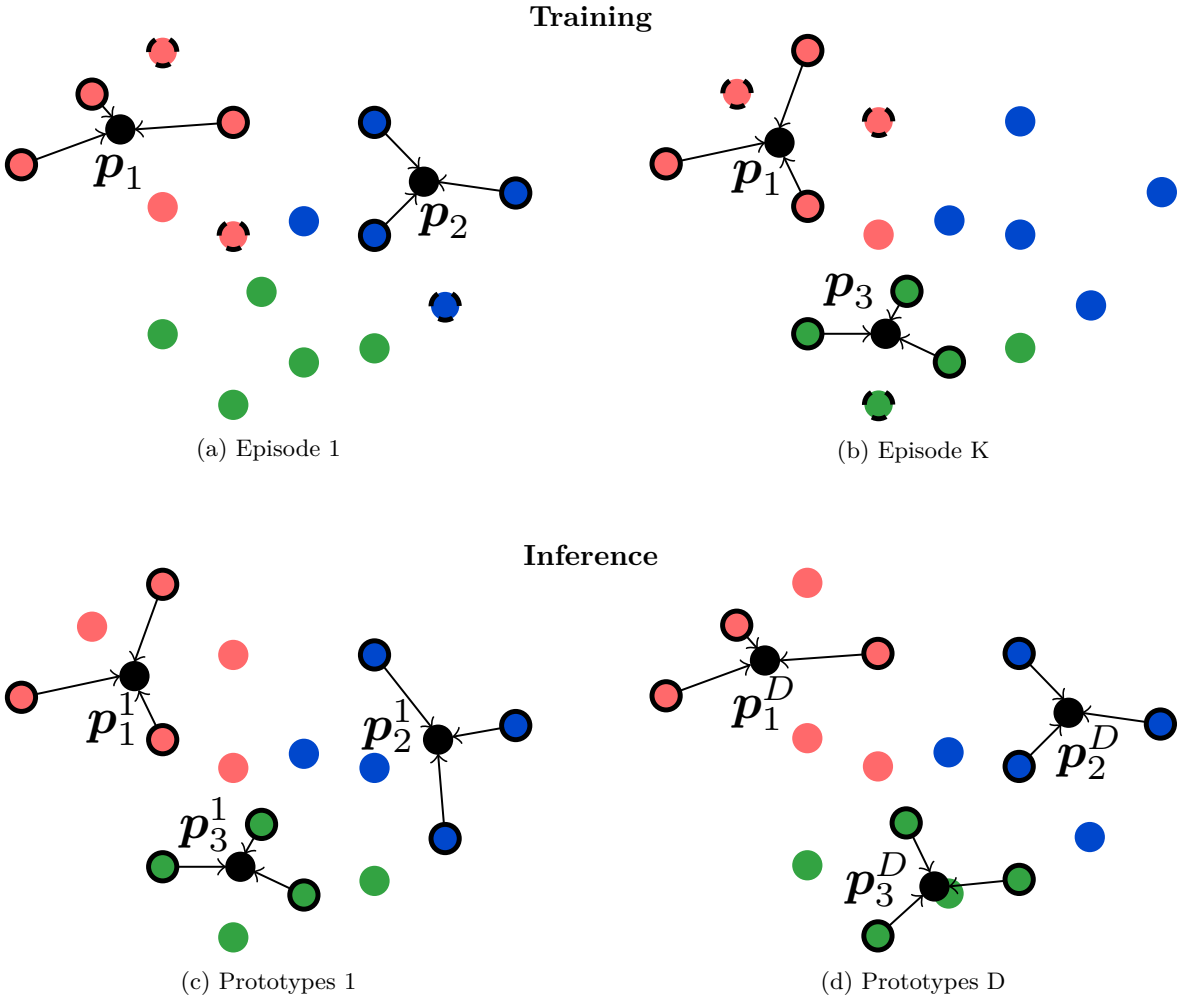
At inference, a sample is assigned to the class whose prototype is the closest to the sample representation, according to a distance d such as the Euclidean distance or the cosine distance:

$$p_{\boldsymbol{\theta}}(y = i|\boldsymbol{x}) = \frac{exp(-d(\pi_{\boldsymbol{\theta}}(\boldsymbol{x}), \boldsymbol{p}_i))}{\sum_j exp(-d(\pi_{\boldsymbol{\theta}}(\boldsymbol{x}), \boldsymbol{p}_j))} \tag{6.5}$$

Prototypical networks are optimized with a meta-learning framework that mimics the inference process during training. At each iteration of the optimization algorithm, called an episode in the framework of prototypical networks, a subset of classes are sampled (leaving some classes aside as if they were not in the training set). From those classes, few random support samples are drawn to build class prototypes (usually as many support examples as labels given at inference on new classes), which are used to predict the classes of a subset of the remaining samples (similar to a random batch in classic stochastic gradient descent). From those samples, called query samples, the cross-entropy is back-propagated to update the prototypical network parameters. Training episodes are illustrated in Fig. 6.11.

3.2.2 Ensembles of prototypical networks

To our knowledge, prototypical networks have been exclusively applied to the task of few-shot learning. However, we argue that their framework can benefit to a conventional classification task. Precisely, we suggest to build an ensemble of classifiers from K different sets of class prototypes. Ensembles of machine learning models are known to be a simple way of improving



(a) Episode 1 (b) Episode K

(c) Prototypes 1 (d) Prototypes D

Figure 6.11: **Top row**: illustration of the optimization of prototypical networks. At each episode, random classes are sampled from which random support examples (dots surrounded by dark circles) are used to compute class prototypes (black dots denoted as $\boldsymbol{p}_i$). Prototypes are used to predict the classes of query samples (dots surrounded by dashed dark circles), randomly sampled from the same classes. **Bottom row**: illustration of the proposed inference procedure. D prototypes are sampled to build an ensemble of classifiers, whose average gives the final prediction.

the performance of individual models [Dietterich, 2000]. Ensemble methods consist in optimizing several models on the same data and averaging their prediction at inference. While training large ensemble of neural networks can be computationally expensive, prototypical networks offer a cheap way to build large ensembles when there are applied on traditional classification tasks (as more than a few samples per class are available at inference). Inference with ensemble of prototypes is illustrated in Fig. 6.11.

3.3 Preliminary experiments

In this section, we experimentally study the benefits of self-supervision combined with a k-nearest neighbors algorithm and a Random Forest classifier against supervised models. Precisely, we tested a Random Forest (RF), a k-nearest neighbors algorithm (KNN), a classic multilayer perceptron (MLP), a 1D CNN [Hu et al., 2015], a 3D CNN [Li et al., 2017] with a 5×5 patch size, a spectral attention model (HSI) [Thoreau et al., 2021] (developed during my internship at ONERA and described in the appendix) as well as p^{3}VAE on the Toulouse data set, in comparison to an Ensemble of Prototypical Networks (EPN), a KNN classification

from representations learned with conventional autoencoders and with masked autoencoders. Further comparison with more sophisticated models will be done in future work.

3.3.1 Hyperparameters

We selected hyperparameters through a random search by testing, for each model and for each split, 20 random combinations.

RF Hyperparameters of the RandomForestClassifier sklearn object[3] are as follows: $n\_estimators \in \{100, 400, 800, 1600, 2400\}$, $min\_samples\_split \in \{2, 5, 10\}$, $min\_samples\_leaf \in \{1, 2, 4\}$, $max\_depth \in \{10, 20, 30, 40, 50, 60, 70, 80, 90, 100, None\}$, $max\_features \in \{auto, sqrt\}$, $bootstrap \in \{True, False\}$. The class weights are inversely proportional to class frequencies.

MLP $lr \in [5e^{-5}, 5e^{-4}]$, $\alpha \in [1e^{-10}, 1e^{-2}]$, $wd \in [1e^{-10}, 1e^{-2}]$, $h_{dim} \in \{64, 512, 2056\}$, where α weighted a L1 norm regularization, wd weighted a L2 norm regularization and h_{dim} was the number of neurons of the 4 hidden layers. Number of parameters ranged from $\approx 34,000$ to $\approx 13,000,000$.

CNN 1D $lr \in [5e^{-5}, 5e^{-4}]$, $wd \in [1e^{-10}, 1e^{-2}]$, $h_{dim} \in \{32, 512, 2056\}$ and $n_{convs} \in \{32, 64, 128\}$ where n_{convs} was the number of kernels of the convolutional layer. Number of parameters range from $\approx 42,000$ to $\approx 10,000,000$.

CNN 3D $lr \in [5e^{-5}, 5e^{-4}]$, $wd \in [1e^{-10}, 1e^{-2}]$, $n_{planes} \in \{2, 32, 128, 256\}$ and n_{planes} was the number of convolution kernels of the first layer. Number of parameters range from $\approx 43,000$ to $\approx 9,000,000$.

p^{3}VAE $lr \in [5e^{-5}, 5e^{-4}]$, $|\boldsymbol{z}| \in \{4, 8, 16\}$, $\lambda_{sam} \in [1e^{-2}, 1]$, $\lambda_{encoder} \in [1e^{-3}, 1e^{-1}]$, $\lambda_{entropy} \in [1e^{-3}, 1]$, $\lambda_{classifier} \in [1e^{-3}, 1e^{-1}]$, $\beta \in [1e^{-5}, 1e^{-3}]$, $\beta_g \in [0, 1]$. Number of parameters ranged from $\approx 775,000$ to $\approx 1,730,000$.

Ensemble of Prototypical Nets (EPN) $lr \in [5e^{-5}, 5e^{-4}]$, $\alpha \in [1e^{-10}, 1e^{-2}]$, $h_{dim} \in \{64, 512, 2056\}$. Number of parameters ranged from $\approx 65,000$ to $\approx 22,000,000$.

HSI $lr \in [1e^{-5}, 1e^{-3}]$, $p_{dropout} \in [0, 1]$, $n_{indices} \in \{100, 500, 1000\}$, $k_{bands} \in \{3, 5, 10\}$ and $\sigma \in \{0.5, 2, 4\}$. Number of parameters ranged from $\approx 3,800$ to $\approx 38,000$.

3.3.2 Results & Discussion

Preliminary quantitative and qualitative results are given in Tab. 6.4 and Fig. 6.13, respectively.

Very light-weight HSI outperformed over-parameterized MLP. On the test set, HSI reached higher performances than MLPs with much fewer parameters and comparable accuracy than CNNs and ensemble of prototypical networks. While the spectral features HSI and MLP can extract should be quite similar, it seems that the architecture of HSI is more suited to learn discriminating features with very few parameters. Another unexpected observation is that better validation performances (OA and F1 score) were usually obtained (approximately half of the time) with over-parameterized MLPs than light-weight MLPs

[3] https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

during the random search.

Non-parametric classification algorithms outperformed supervised models. Non-parametric RF and KNN significantly outperformed every parametric models. RF in particular reached a higher higher F1 score by more than 10% compared to parametric models.

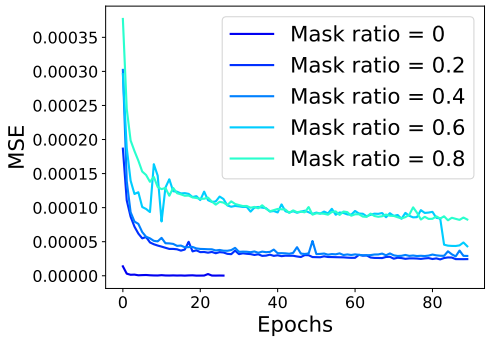
p^{3}VAE did not converge on the training set. We did not succeed to make the objective function of p^{3}VAE converge on the training set, resulting in a low accuracy on the training data, and obviously even worse accuracy on the test data. It seems that p^{3}VAE learned spurious correlations between the spectra and irradiance conditions, which induced many confusions including confusions between easily separable classes. We believe that the training did not converge because of the strong *intrinsic* intra-class spectral variations compared to little *physics* intra-class spectral variations between the labeled and unlabeled training sets. In order to reduce the impact of misleading unlabeled data, we optimized p^{3}VAE in a supervised way only, without getting significantly better results as the labeled data probably did not represent enough different irradiance conditions.

Unsupervised / self-supervised learning combined with random forests is a strong baseline. RF applied on the latent space of a conventional autoencoder with dense layers outperformed a RF on raw spectra by margins of 7% and 9% in terms of OA and F1 score, respectively. RF applied on the latent space (precisely [CLS] tokens) of the MAE reached only better OA and F1 score by 3% than a raw RF. We believe that the lower accuracy reached with the MAE compared to a conventional AE may be due to the transformer architecture used for the masked reconstruction task which we do not think to be appropriate for hyperspectral data, as discussed previously.

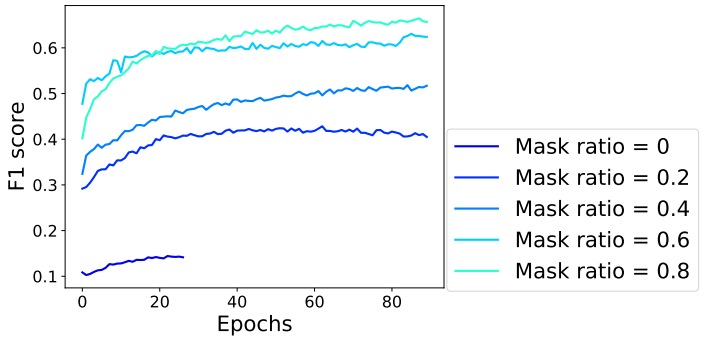
AE / MAE + RF fails in the shadows and on mixed pixels Qualitatively, land cover maps in Fig. 6.13 show that the class of many pixels in shadows are not well predicted. For instance, the road in *asphalt* around the *Grand Rond* is confused with *slate* in the shadows of trees, and pixels at the edge of *trees* over *water* are predicted as *seaweed* in the *Cale de Radoub*.

Masking ratio Fig. 6.12 shows a preliminary ablation study about the masking ratio. We studied how the ratio influenced the reconstruction error and the classification accuracy (obtained with a Random Forest) on the validation set, all other hyperparameters being equal (including the weights initialization). Though more experiments with different weights initialization should be made to validate our findings, the experiments are consistent with our a priori intuition: a low masking ratio makes the task too trivial, which results in a very high reconstruction error but a low F1 score. In contrast, a high masking ratio results in a higher reconstruction error but a higher F1 score. Those results suggest that the autoencoder learns more discriminating spectral representations when the task is difficult enough, that is when the autoencoder is compelled to learn informative correlations between spectral features rather than just "filling the holes".

Training time While the inference time of supervised and unsupervised methods is barely the same (as it depends only on the model size), the training time scales with the size of the data set, which is 14× larger for unsupervised techniques for the Toulouse data set.



(a) Reconstruction error



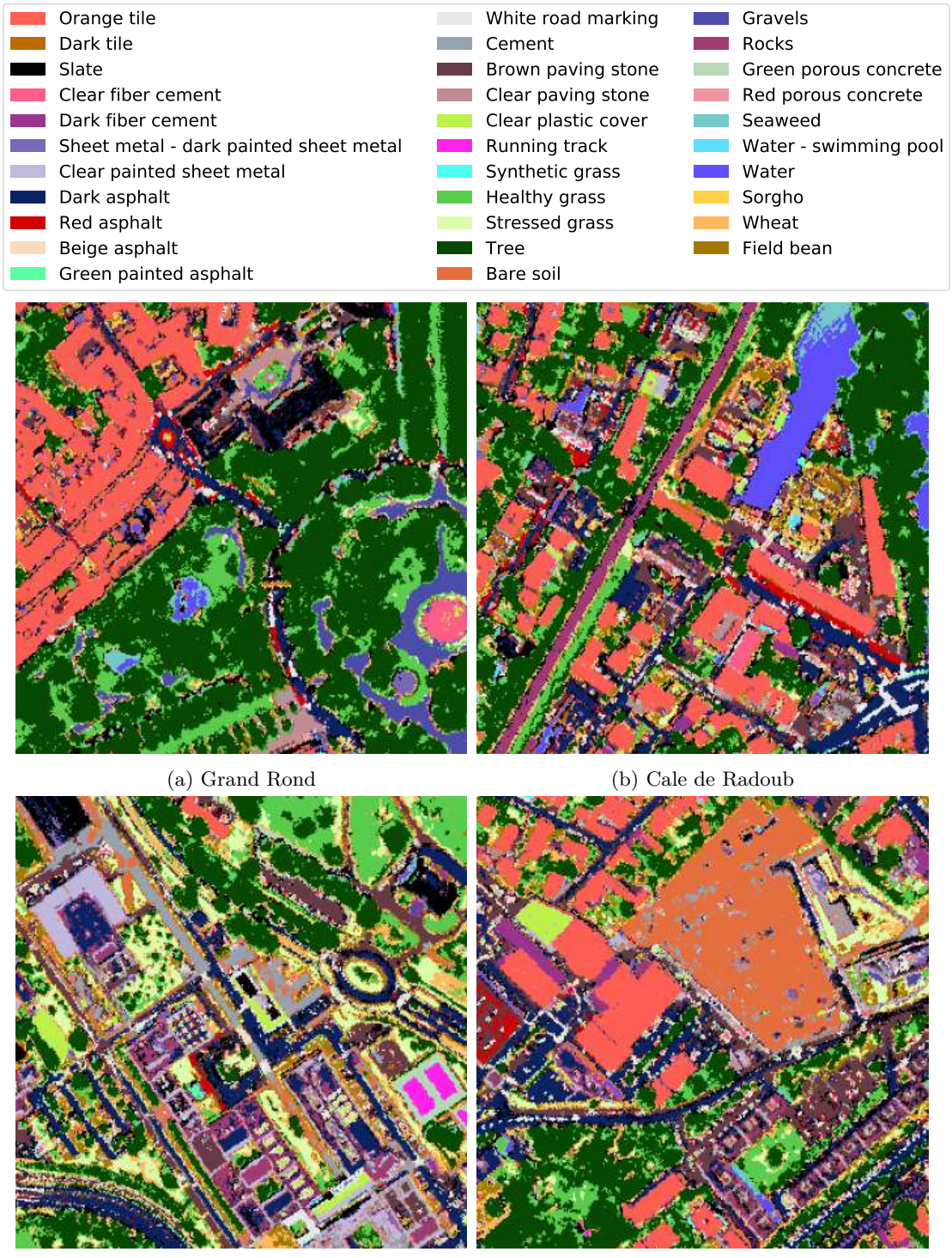
(b) F1 score

Figure 6.12: Metrics on the validation set for different masking ratio

Table 6.4: Average overall accuracy and F1 score over 8 splits

	Model	OA	F1 score
Supervised	*Parametric*		
	MLP	0.64	0.50
	CNN 1D	0.66	0.53
	CNN 3D	0.68	0.53
	HSI	0.69	0.53
	Parametric representation + non-parametric classification		
	EPN	0.69	0.53
	Non-parametric		
	KNN	0.71	0.60
	RF	0.75	0.65
Semi-supervised	p^3VAE	$\emptyset^1$	$\emptyset^1$
	AE + KNN	0.77	0.67
	MAE + KNN	0.73	0.61
	MAE + RF	0.78	0.68
	AE + RF	**0.82**	**0.74**

[1] We do not report the metrics of p^3VAE on the test set because convergence was far from being achieved on the training set.



(a) Grand Rond

(b) Cale de Radoub

(c) ISAE-SUPAERO

(d) Roseraie

Figure 6.13: Land cover maps produced from the worst AE + KNN classifier over the 8 splits

4 Conclusions and perspectives

4.1 Conclusions

We introduced the Toulouse Hyperspectral Data Set, a new data set that mainly stands out for its size, its larger spectral variability, its larger land cover and land use nomenclatures and above all for its standard train / test sets compared to public hyperspectral data sets. We believe that it raises new questions and hope that it will open new research directions in hyperspectral image analysis such as multi-label segmentation, long-tailed class imbalance learning, hierarchical segmentation and semi-supervised learning. The hyperspectral data are already available at https://camcatt.sedoo.fr/ and we will release the ground truth soon with a python package to easily build Pytorch loaders.

Our preliminary experiments have shown that parametric models are prone to over-fitting despite regularization techniques, while the simple non-parametric k-nearest neighbors algorithm demonstrated high performances. While we expected the masked reconstruction task to learn a latent space suited for classification, a conventional reconstruction task led to better results, though we believe that further experiments are worth considering. Finally, we could not experiment the combination of self-supervision with prototypes because of lack of time. However, we believe that Prototypical Networks are promising, as they combine the representation power of neural networks with a non-parametric classification scheme and interesting ensemble properties.

4.2 Perspectives

■ **Masked Autoencoder without a transformer architecture**. Whether the masked reconstruction task could be done with more appropriate neural network architectures than transformers is, to our opinion, a serious research avenue for self-supervised representation learning of hyperspectral data.

■ **Semi-supervised and self-supervised learning**. In future work, we would like to combine semi-supervised and self-supervised learning. First, we would like to simply add a supervised loss to the unsupervised (respectively self-supervised) loss of autoencoders (respectively masked autoencoders). Then, we believe that masked autoencoders could be used along side a clustering-based self-supervised technique guided with class prototypes learned on the labeled data. In this framework, we could also consider physically meaningful data augmentation techniques on the spectral dimension. The difficulty is that, without knowing the true local irradiance condition of a pixel, simulating spectral variations induced by variations of the irradiance conditions is not possible.

■ **Multi-label segmentation**. Jointly learning contextual features for land use segmentation and spectral features for land cover segmentation may be an interesting research path. Firstly, some land use and land cover classes are correlated, such as *roofs* and *tile* or *cultivated field* and *wheat*,which could help to raise ambiguities. Second, a land cover nomenclature will never be exhaustive enough to represent every materials in an urban area. In contrast, a land use nomenclature can easily describe a urban landscape with a few dozen classes. Therefore, a multi-label segmentation with additional land use predictions could be helpful when the uncertainty on the land cover is high.

■ **Long-tailed class imbalance**. The problem of long-tailed class distribution has been poorly studied in the hyperspectral community, though it is a key issue to develop operational

segmentation models. In particular, a question that is little discussed, in the machine learning community as well, is how unsupervised techniques can circumvent heavy class imbalance. As a matter of fact, simple methods can balance the influence of large classes on supervised losses, but balancing the impact of those classes on unsupervised losses remains an open question.

■ **Hierarchical segmentation**. Recent works, for instance [Landrieu and Garnot, 2021], introduced supervised optimization techniques to leverage the class hierarchy, considering that all mistakes are not equal. In the context of impermeable vs permeable surfaces segmentation, such techniques may be very useful and the nomenclature of the Toulouse data set now allows to investigate those techniques. It also opens the path towards the research area of Hierarchical Multi-Label Classification which consists in the classification of instances that may belong to multiple classes that organize in a hierarchy [Cerri et al., 2014; Vens et al., 2008; Wehrmann et al., 2018].

The work presented in this chapter has resulted in a publication submitter to ISPRS Journal of Photogrammetry and Remote Sensing:

- R. Thoreau, L. Risser, V. Achard, B. Berthelot and X. Briottet, "Toulouse Hyperspectral Data Set: a benchmark data set to assess semi-supervised spectral representation learning and pixel-wise classification techniques," arXiv preprint arxiv.org/abs/2311.08863.

5 Appendices

5.1 Learning hyperspectral indices through soft attention [Thoreau et al., 2021]

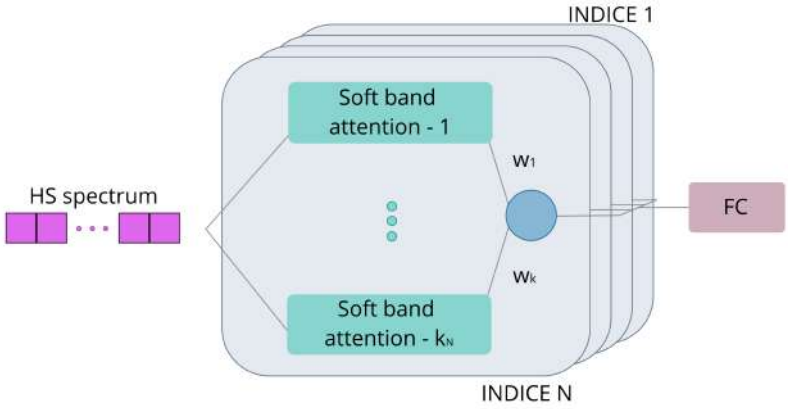


Figure 6.14: Schematic view of our HSI learning model

We denote the reflectance spectrum of a pixel by $\boldsymbol{x} \in [0,1]^B$ where B is the number of spectral bands. Each indice I_n (n $\in \{1,...,N\}$) has k_n soft-attention blocks that focus on the neighborhood of specific bands. Each soft-attention block computes a vector $\mathbf{a_i} \in \mathbb{R}^B$ (i $\in \{1,...,k_n\}$) defined by its mean (trainable parameter) and its standard deviation (hyperparameter). Finally, indices are a linear combination of the selected bands, thresholded by a ReLU function. The motivation behind this design is that the band attention mechanism is differentiable so that the model is trainable by a conventional gradient descent algorithm with regard to the cross-entropy loss. The HSI learning unit can be summarized by the following equations:

$$I_n(s) = ReLU\Big(\sum_{i=1}^{k_n} w_i(\mathbf{a_i}^T\mathbf{s})\Big) \quad \forall n \in \{1,...,N\} \tag{6.6}$$

For a given indice I_n, for all $b \in \{1,...,B\}$ and for all $i \in \{1,...,k_n\}$:

$$a_i^b = \frac{1}{\sigma\sqrt{2\pi}} e^{-(b-\mu_i)^2/2\sigma^2} \tag{6.7}$$

where σ and μ_i are the standard deviation and the mean of the soft band attention unit. The output of the HSI units is a N-dimensionnal vector that feeds one fully-connected layer. To sum up, the parameters of the model are $\boldsymbol{\theta} = \{\boldsymbol{\mu}, \mathbf{W}, \boldsymbol{\theta_f}\}$ where $\boldsymbol{\mu}$ is the matrix of the bands attention, $\boldsymbol{W}$ is the matrix of their associated weights and $\boldsymbol{\theta_f}$ are the parameters of the fully-connected (FC) layer. The complete architecture of the model is illustrated in figure 6.14.

6 References

Audebert, N., Le Saux, B., and Lefèvre, S. (2019). Deep learning for classification of hyperspectral data: A comparative review. *IEEE geoscience and remote sensing magazine*, 7(2):159–173. 162

Caron, M., Bojanowski, P., Joulin, A., and Douze, M. (2018). Deep clustering for unsupervised learning of visual features. In *Proceedings of the European conference on computer vision (ECCV)*, pages 132–149. 168

Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., and Joulin, A. (2020). Unsupervised learning of visual features by contrasting cluster assignments. *Advances in neural information processing systems*, 33:9912–9924. 168

Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., and Joulin, A. (2021). Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660. 168

Cerri, R., Barros, R. C., and De Carvalho, A. C. (2014). Hierarchical multi-label classification using local neural networks. *Journal of Computer and System Sciences*, 80(1):39–56. 177

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. (2020). A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR. 168

Dietterich, T. G. (2000). Ensemble methods in machine learning. In *International workshop on multiple classifier systems*, pages 1–15. Springer. 171

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. 169

Dosovitskiy, A., Springenberg, J. T., Riedmiller, M., and Brox, T. (2014). Discriminative unsupervised feature learning with convolutional neural networks. *Advances in neural information processing systems*, 27. 168

Duan, P., Xie, Z., Kang, X., and Li, S. (2022). Self-supervised learning-based oil spill detection of hyperspectral images. *Science China Technological Sciences*, 65(4):793–801. 168

Fini, E., Astolfi, P., Alahari, K., Alameda-Pineda, X., Mairal, J., Nabi, M., and Ricci, E. (2023). Semi-supervised learning made simple with self-supervised clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3187–3197. 168

Geiß, C., Pelizari, P. A., Schrade, H., Brenning, A., and Taubenböck, H. (2017). On the effect of spatially non-disjoint training and test samples on estimated model generalization capabilities in supervised classification with spatial features. *IEEE Geoscience and Remote Sensing Letters*, 14(11):2008–2012. 162

Gidaris, S., Singh, P., and Komodakis, N. (2018). Unsupervised representation learning by predicting image rotations. In *ICLR 2018*. 168

He, K., Chen, X., Xie, S., Li, Y., Dollár, P., and Girshick, R. (2022). Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009. xiii, 157, 168, 169, 170

Henaff, O. (2020). Data-efficient image recognition with contrastive predictive coding. In *International conference on machine learning*, pages 4182–4192. PMLR. 168

Hinton, G., Vinyals, O., and Dean, J. (2015). Distilling the knowledge in a neural network. In *NIPS Deep Learning and Representation Learning Workshop*. 168

Hong, D., Han, Z., Yao, J., Gao, L., Zhang, B., Plaza, A., and Chanussot, J. (2021). Spectralformer: Rethinking hyperspectral image classification with transformers. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–15. 169

Hu, W., Huang, Y., Wei, L., Zhang, F., and Li, H. (2015). Deep convolutional neural networks for hyperspectral image classification. *Journal of Sensors*, 2015:1–12. 171

Landrieu, L. and Garnot, V. S. F. (2021). Leveraging class hierarchies with metric-guided prototype learning. In *British Machine Vision Conference (BMVC)*. 177

Lange, J., Cavallaro, G., Götz, M., Erlingsson, E., and Riedel, M. (2018). The influence of sampling methods on pixel-wise hyperspectral image classification with 3d convolutional neural networks. In *IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium*, pages 2087–2090. IEEE. 162

Li, Y., Zhang, H., and Shen, Q. (2017). Spectral–spatial classification of hyperspectral imagery with 3d convolutional neural network. *Remote Sensing*, 9(1):67. 171

Oord, A. v. d., Li, Y., and Vinyals, O. (2018). Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*. 168

Qian, Y., Zhu, H., Chen, L., and Zhou, J. (2022). Hyperspectral image restoration with self-supervised learning: A two-stage training approach. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–17. 168

Qin, Y., Ye, Y., Zhao, Y., Wu, J., Zhang, H., Cheng, K., and Li, K. (2023). Nearest neighboring self-supervised learning for hyperspectral image classification. *Remote Sensing*, 15(6). 168

Roupioz, L., Briottet, X., Adeline, K., Al Bitar, A., Barbon-Dubosc, D., Barda-Chatain, R., Barillot, P., Bridier, S., Carroll, E., Cassante, C., Cerbelaud, A., Déliot, P., Doublet, P., Dupouy, P., Gadal, S., Guernouti, S., De Guilhem De Lataillade, A., Lemonsu, A., Llorens, R., Luhahe, R., Michel, A., Moussous, A., Musy, M., Nerry, F., Poutier, L., Rodler, A., Riviere, N., Riviere, T., Roujean, J., Roy, A., Schilling, A., Skokovic, D., and Sobrino, J. (2023). Multi-source datasets acquired over toulouse (france) in 2021 for urban microclimate studies during the camcatt/ai4geo field campaign. *Data in Brief*, 48:109109. 157, 158

Snell, J., Swersky, K., and Zemel, R. (2017). Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30. 157, 170

Thoreau, R., Achard, V., and Briottet, X. (2021). Hyperspectral classification based on spectral indices learned through soft attention units. In *2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS*, pages 2544–2547. 155, 171, 178

Tian, Y., Krishnan, D., and Isola, P. (2020). Contrastive multiview coding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*, pages 776–794. Springer. 168

Van der Maaten, L. and Hinton, G. (2008). Visualizing data using t-sne. *Journal of machine learning research*, 9(11). 163

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30. 169

Vens, C., Struyf, J., Schietgat, L., Džeroski, S., and Blockeel, H. (2008). Decision trees for hierarchical multi-label classification. *Machine learning*, 73:185–214. 177

Wehrmann, J., Cerri, R., and Barros, R. (2018). Hierarchical multi-label classification networks. In *International conference on machine learning*, pages 5075–5084. PMLR. 177

Wu, Z., Xiong, Y., Yu, S. X., and Lin, D. (2018). Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3733–3742. 168

Zhai, X., Oliver, A., Kolesnikov, A., and Beyer, L. (2019). S4l: Self-supervised semi-supervised learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1476–1485. 168

Zhang, Y., Kang, B., Hooi, B., Yan, S., and Feng, J. (2023). Deep long-tailed learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 164

Zhao, L., Luo, W., Liao, Q., Chen, S., and Wu, J. (2022). Hyperspectral image classification with contrastive self-supervised learning under limited labeled samples. *IEEE Geoscience and Remote Sensing Letters*, 19:1–5. 168

Zhu, L., Wu, J., Biao, W., Liao, Y., and Gu, D. (2023). Spectralmae: Spectral masked autoencoder for hyperspectral remote sensing image reconstruction. *Sensors*, 23(7):3728. 169

Chapter 7

Conclusions and perspectives

1 General conclusions

We believe that this thesis is a step towards the automatic mapping of impermeable surfaces in metropolises from airborne hyperspectral images thanks to three main contributions:

- **A global methodology to improve the training data set** In chapter 4, we demonstrated that Active Learning techniques, eventually combined with our preprocessing techniques, can significantly help engineers and researchers during field campaigns to 1) explore the hyperspectral image and discover rare but important land cover classes with BALD [Houlsby et al., 2011], 2) annotate pixels that are particularly informative of the differences between similar classes with Breaking Ties [Tong Luo et al., 2004] or our enhanced version Probabilistic Breaking Ties and 3) select few pixels to dramatically increase the representativeness of the training data with Core-set [Sener and Savarese, 2018]. The potential of these methods have been quantitatively outlined in an operational scenario by significantly increasing the semantic segmentation performances (with at least +10% accuracy with only 300 additional labeled pixels compared to a random sampling). Notwithstanding, one could argue that a human expert would not randomly select pixels but would rather annotate specific samples according to some intuition acquired from experience. While we did not evaluate the performance of human experts, we are convinced that AL algorithms have an unmatched capacity to process very large amounts of data much more rapidly and to select highly informative samples. Yet, to benefit from the full potential of AL methods, we argue that engineering efforts are required to design convenient tools that would help engineers and researchers during field campaigns.

- **A hybrid model robust to local variations of illumination** In chapter 5, we demonstrated that the combination of a simplified radiative transfer model with a generative model can significantly outperform the extrapolation performances of a machine learning model alone on spectra with irradiance conditions that were not seen during training. Our hybrid model also showcased interesting disentanglement and interpretability properties. While it may appear obvious, we believe that latent generative models are particularly suited to integrate physical models. In cases where physics can describe a large part of the data variability, our experiments demonstrated that our hybrid model can reach a good local optima through stochastic gradient descent. In other situations though, the optimization was often not successful and the model learned spurious correlations between illumination conditions and data.

- **A very large hyperspectral data set** In chapter 6, we introduced a new hyperspectral data set over the city of Toulouse, France. We qualitatively demonstrated that the

Toulouse data set is more appropriate to evaluate the generalization performances of semantic segmentation models, especially of unsupervised and semi-supervised methods, thanks to standard training, validation and test sets, as well as a higher complexity compared to other public hyperspectral data sets. On this data set, spectral representations learned from an auto-encoder and a masked auto-encoder combined with a random forest provide a strong baseline for land cover classification, which is far superior to the results obtained with parametric models optimized with class supervision.

Overall, we believe that a simple yet important conclusion of our work is that **land cover classes are non-abstract classes that need features of low abstraction in contrast to (abstract) land use classes that need features of high abstraction**. That being said, spatial information could raise ambiguities about the land cover in certain cases, and we open up a few perspectives in this direction in what follows.

2 General perspectives

■ **Active Learning with few-shot and self-supervised learning?** While we experimented Active Learning techniques with a spectral CNN, non-parametric classification techniques combined with spectral representation learning may be more relevant. A key issue in AL is indeed to learn significantly different decision boundaries with few additional labeled pixels only.

■ **Few-shot hybrid modeling?** We tried with p^{3}VAE to integrate deductive biases in a machine learning model. Our optimization technique though still required adequate labeled and unlabeled data to properly converge. Whether deductive biases can be integrated in a few-shot manner is, in our opinion, a question worth studying.

■ **Learning from larger spatial contexts?** In the literature, most spatial-spectral methods still focus on pixel-wise classification from relatively small patches (less than 30×30 pixel patches for about a 1 meter GSD) which we believe to be not enough to contain contextual information. We argue that a key issue is to handle large patches, at least of 64×64 pixels approximately, with fully-connected convolutional neural networks, in order to learn from spatial context (state-of-the-art semantic segmentation techniques on very high resolution RGB satellite images usually process 256×256 pixel wide images). Processing such large hyperspectral patches with 3D convolutional neural networks however raises memory issues. The distinct optimization of spectral methods to learn low dimensional spectral representations and spatial CNNs may be a promising research path.

■ **Supervision with large land use data sets?** Several land use databases are publicly available, such as Open Street Map[1] or Urban Atlas[2], that may be used for pre-training segmentation models before fine-tuning on the land use ground truth of the Toulouse Hyperspectral data set.

■ **Combination with vision foundation models?** Intensive research efforts have been conducted in the field of semantic segmentation to cope with the spatial intra-class variability. As a matter of fact, most discriminative information of common RGB images is textural, geometric and contextual information. Recently, the vision transformer (ViT) introduced in [Dosovitskiy et al., 2020] has opened the path towards powerful models for computer vision

[1] https://www.openstreetmap.fr/
[2] https://land.copernicus.eu/local/urban-atlas

tasks, with the ability to leverage wide spatial contexts, at the cost however of long pre-training time on very large data sets [Dosovitskiy et al., 2020; Strudel et al., 2021]. Various works have extended the ViT to semantic segmentation like in [Gu et al., 2022; Strudel et al., 2021] or in [Zheng et al., 2021] by combining the transformer with convolutions. Very recently, [Kirillov et al., 2023] introduced a foundation model, namely SAM, for computer vision, *i.e.* a model trained on diverse and very large data that can generalize to new tasks and out-of-distribution data. The generalization capacities of SAM rely on prompts, that are points, boxes or masks that specify which objects in the image to segment. Combining this foundation model with more specialized models that can feed SAM with prompts is a very promising way to perform zero-shot transfer learning on various tasks such as instance segmentation, though using SAM for semantic segmentation remains unclear Kirillov et al. [2023]. Therefore, there is a hot topic to tackle that is the adaptation of SAM to semantic segmentation with the integration of spectral models.

3 References

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*. 184, 185

Gu, J., Kwon, H., Wang, D., Ye, W., Li, M., Chen, Y.-H., Lai, L., Chandra, V., and Pan, D. Z. (2022). Multi-scale high-resolution vision transformer for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12094–12103. 185

Houlsby, N., Huszár, F., Ghahramani, Z., and Lengyel, M. (2011). Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*. 183

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., Dollár, P., and Girshick, R. (2023). Segment anything. *arXiv preprint arXiv:2304.02643*. 185

Sener, O. and Savarese, S. (2018). Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*. 183

Strudel, R., Garcia, R., Laptev, I., and Schmid, C. (2021). Segmenter: Transformer for semantic segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7262–7272. 185

Tong Luo, Kramer, K., Samson, S., Remsen, A., Goldgof, D. B., Hall, L. O., and Hopkins, T. (2004). Active learning to recognize multiple types of plankton. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, volume 3, pages 478–481 Vol.3. 183

Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., Fu, Y., Feng, J., Xiang, T., Torr, P. H., et al. (2021). Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6881–6890. 185

Conclusion en français

Conclusions générales

Nous pensons que cette thèse constitue une étape vers la cartographie automatique des surfaces imperméabilisées dans les milieux urbains à partir d'images hyperspectrales aéroportées grâce à trois contributions principales :

- **Une méthodologie globale pour améliorer la qualité des bases d'apprentissage** Dans le chapitre 4, nous avons démontré que les techniques d'*Active Learning*, éventuellement combinées à nos techniques de prétraitement, peuvent aider considérablement les ingénieurs et les chercheurs lors des campagnes terrain à 1) explorer l'image hyperspectrale et découvrir des classes d'occupation du sol rares mais importantes avec BALD [Houlsby et al., 2011], 2) annoter les pixels qui sont particulièrement informatifs sur les différences entre des classes similaires avec Breaking Ties [Tong Luo et al., 2004] ou notre version étendue Probabilistic Breaking Ties et 3) sélectionner quelques pixels pour augmenter considérablement la représentativité des données d'entraînement avec Coreset [Sener and Savarese, 2018]. Le potentiel de ces méthodes a été quantitativement mis en évidence dans un scénario opérationnel en augmentant de manière significative les performances de segmentation sémantique (avec au moins +10% de précision avec seulement 300 pixels annotés supplémentaires par rapport à un échantillonnage aléatoire). Néanmoins, si un expert humain ne sélectionnerait pas les pixels au hasard, mais annoterait plutôt certains pixels qu'il jugerait informatifs de par son expérience, nous sommes convaincus que les algorithmes d'AL ont une capacité inégalée à traiter de très grandes quantités de données beaucoup plus rapidement et à sélectionner des échantillons hautement informatifs. Cependant, pour bénéficier du plein potentiel de l'*Active Learning*, nous soutenons que des efforts d'ingénierie sont nécessaires pour concevoir des outils pratiques qui aideraient les ingénieurs et les chercheurs pendant les campagnes sur le terrain.

- **Un modèle hybride robuste aux variations locales d'éclairement** Dans le chapitre 5, nous avons démontré que la combinaison d'un modèle de transfert radiatif simplifié et d'un modèle génératif peut dépasser de manière significative les performances d'extrapolation d'un modèle d'apprentissage automatique conventionnel pour la classification de matériaux dont les conditions d'éclairement n'ont pas été observées pendant l'entraînement. Notre modèle hybride présente également des propriétés intéressantes de désenchevêtrement et d'interprétabilité. Bien que cela puisse paraître évident, nous pensons que les modèles génératifs latents sont particulièrement adaptés à l'intégration de modèles physiques. Dans les cas où les variations *physiques* décrivent une grande partie de la variabilité des données, nos expériences ont démontré que notre modèle hybride peut être optimisé efficacement par descente de gradient stochastique. Cependant, lorsque la variabilité intra-classe est principalement *intrinsèque* et *sémantique*, l'optimisation peine à converger et le modèle apprend des corrélations fallacieuses entre les conditions d'éclairement et les spectres.

■ **Une très grande base de données hyperspectrales** Dans le chapitre 6, nous avons introduit une nouvelle base de données hyperspectrales sur la ville de Toulouse, France. Nous avons démontré qualitativement que le jeu de données de Toulouse est plus approprié pour évaluer les performances de généralisation des modèles de segmentation sémantique, en particulier des méthodes non supervisées et semi-supervisées, grâce à des ensembles d'entraînement, de validation et de test standards, ainsi qu'à une plus grande complexité par rapport à d'autres jeux de données hyperspectrales de la littérature. Sur ces données, un apprentissage de représentations spectrales à partir d'un auto-encodeur et d'un auto-encodeur masqué combiné avec une forêt aléatoire constitue un premier résultat de référence, bien supérieur aux résultats obtenus avec des modèles paramétriques optimisés de manière supervisée.

Dans l'ensemble, nous pensons qu'une conclusion peut-être naïve mais importante de notre travail est que **les classes d'occupation des sols sont des classes non abstraites qui ont besoin de représentations (spectrales) de faible abstraction contrairement aux classes d'usage des sols qui sont abstraites et ont ainsi besoin de représentations (spatiales) de haute abstraction.**

Cela étant dit, les informations spatiales pourraient soulever des ambiguïtés sur l'occupation des sols dans certains cas, et nous ouvrons quelques perspectives dans ce sens dans ce qui suit.

Perspectives générales

■ **Combiner l'*Active Learning* avec le *few-shot learning* et le *self-supervised learning* ?** Bien que nous ayons expérimenté les techniques d'*Active Learning* avec un CNN spectral, les techniques de classification non paramétriques combinées à l'apprentissage de représentations spectrales pourraient être plus pertinentes. L'un des principaux problèmes de l'*Active Learning* est en effet d'apprendre des frontières de classification significativement différentes avec seulement quelques pixels annotés supplémentaires.

■ **La modélisation hybride dans un cadre de *few-shot learning* ?** La technique d'optimisation de p^3VAE, qui intègre des biais déductifs, nécessite des données annotés et non annotées adéquates pour converger correctement. La question de savoir si ces biais déductifs peuvent être intégrés avec encore moins de données mérite à notre avis d'être étudiée.

■ **Apprendre à partir de plus grands contextes spatiaux ?** Dans la littérature, la plupart des méthodes d'apprentissage de représentations spatiales-spectrales se concentrent encore sur la classification par pixel à partir d'imagettes relativement petites (moins de 30×30 pixels pour une résolution spatiale d'environ 1 m), ce qui, selon nous, n'est pas suffisant pour contenir des informations contextuelles. Nous pensons qu'il est essentiel de traiter des imagettes de grande taille, au moins de 64×64 pixels environ, avec des réseaux de neurones convolutionnels entièrement connectés, afin de tirer de l'information du contexte spatial (les techniques de segmentation sémantique de l'état de l'art pour des images satellite RVB à très haute résolution spatiale traitent généralement des images d'une taille de 256×256 pixels). Le traitement de ces grandes imagettes hyperspectrales avec des réseaux de neurones convolutionnels 3D pose toutefois des problèmes de besoin en mémoire. L'optimisation distincte de méthodes spectrales pour apprendre des représentations spectrales de faible dimension et de méthodes spatiales pourrait être une voie de recherche prometteuse.

■ **Supervision avec de grandes bases de données de l'usage des sols?** Plusieurs bases

de données de l'usage des sols sont publiques, comme Open Street Map[3] ou Urban Atlas[4], qui pourraient être utilisées pour pré-entrainer des modèles de sementation avant d'être affiner sur des bases de données d'occupation des sols moins volumineuses.

■ **Combinaison avec des modèles de fondation?** Des recherches intensives ont été menées dans le domaine de la segmentation sémantique pour faire face à la variabilité spatiale intra-classe. En fait, la plupart des informations discriminantes des images RVB courantes sont des informations texturales, géométriques et contextuelles. Récemment, le *vision transformer* (ViT) introduit dans [Dosovitskiy et al., 2021] a ouvert la voie à des modèles puissants pour les tâches de vision par ordinateur, avec la capacité d'exploiter de vastes contextes spatiaux, au prix toutefois d'un long temps de pré-apprentissage sur de très grands ensembles de données [Dosovitskiy et al., 2021; Strudel et al., 2021]. Plusieurs travaux ont étendu le ViT à la segmentation sémantique, comme dans [Gu et al., 2022; Strudel et al., 2021] ou dans [Zheng et al., 2021], en combinant le *transformer* avec des convolutions. Très récemment, [Kirillov et al., 2023] a introduit SAM, un modèle de fondation pour la vision par ordinateur, *i.e.* un modèle optimisé sur des données très volumineuses et variées qui peut se généraliser à de nouvelles tâches et à des données différentes. Les capacités de généralisation de SAM reposent sur des *prompts*, c'est-à-dire des points, des rectangles ou des masques qui spécifient les objets de l'image à segmenter. La combinaison de ce modèle de fondation avec des modèles plus spécialisés qui peuvent alimenter SAM avec des *prompts* est un moyen très prometteur de transférer l'apprentissage sur diverses tâches telles que la segmentation d'instances, bien que l'utilisation de SAM pour la segmentation sémantique reste une question ouverte Kirillov et al. [2023]. Il y a donc fort enjeu à combiner le modèle SAM avec des modèles spectraux pour la segmentation sémantique de l'occupation des sols.

[3] https://www.openstreetmap.fr/
[4] https://land.copernicus.eu/local/urban-atlas