Interactive semantic segmentation of aerial images with deep neural networks

Gaston Lenczner

université PARIS-SACLAY

Interactive semantic segmentation of aerial images with deep neural networks

Segmentation sémantique interactive d'images aériennes avec des réseaux de neurones profonds

Thèse de doctorat de l'université Paris-Saclay

École doctorale n° 580, sciences et technologies de l'information et de la communication (STIC)
Spécialité de doctorat : Traitement du signal et des images
Graduate School : Informatique et sciences du numérique. Référent : Faculté des sciences d'Orsay.

Thèse préparée dans l'unité de recherche Traitement de l'information et systèmes (Université Paris-Saclay, ONERA DTIS/IVA) sous la direction de Dr. Guy LE BESNERAIS (Directeur de recherches, ONERA) et de Dr. Bertrand LE SAUX (Chercheur, ESA/ESRIN) et l'encadrement de Dr. Adrien CHAN-HON-TONG (Chercheur, ONERA) et de Dr. Nicola LUMINARI (Head of data sciences, Alteia).

Thèse soutenue à Paris-Saclay, le 20 septembre 2022, par

Gaston LENCZNER

Composition du jury

Céline HUDELOT Professeur, CentraleSupelec	Présidente
Begüm DEMIR Professeur, Technische Universität Berlin	Rapporteur & Examinatrice
David PICARD Professeur, École des Ponts ParisTech	Rapporteur & Examinateur
Charlotte PELLETIER Maîtresse de conférences, Université Bretagne Sud	Examinatrice
Devis TUIA Professeur associé, École Polytechnique Fédérale de Lausanne	Examinateur
Guy LE BESNERAIS Directeur de recherches, ONERA	Directeur de thèse

THESE DE DOCTORAT
NNT : 2022UPASG067

université PARIS-SACLAY

ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre : Segmentation sémantique interactive d'images aériennes avec des réseaux de neurones profonds
Mots clés : Segmentation sémantique, Interactivité, Humain dans la boucle, Images aériennes

Résumé :

Nous proposons dans cette thèse de mettre en place une collaboration entre un réseau de neurones profond et un utilisateur pour collecter rapidement des cartes de segmentation sémantiques précises d'images de télédétection. En bref, l'utilisateur interagit de manière itérative avec le réseau pour corriger ses prédictions initialement erronées. Concrètement, ces interactions sont des annotations représentant les labels sémantiques. Nos contributions se décomposent en quatre parties.

Premièrement, nous proposons deux schémas d'apprentissage interactif pour intégrer les entrées de l'utilisateur dans les réseaux de neurones profonds. Le premier concatène les annotations de l'utilisateur avec les autres entrées du réseau (comme l'image RGB). Nous l'appliquons à la fois aux architectures convolutionnelles et aux Transformers. La seconde utilise les annotations comme une vérité terrain partielle pour ré-entraîner le réseau. Ensuite, nous proposons une stratégie d'apprentissage actif pour guider l'utilisateur vers les zones les plus pertinentes à annoter. Dans ce but, nous adaptons différentes fonctions d'acquisition issues de l'état de l'art pour évaluer l'incertitude du réseau de neurones. Enfin, nous proposons de modifier l'espace de sortie de l'algorithme pour l'adapter rapidement à de nouvelles classes sous faible supervision. Pour atténuer les problèmes de décalage de la classe d'arrière plan et d'oubli catastrophique inhérents à ce problème, nous comparons différentes régularisations et tirons parti d'une stratégie dite de pseudo-labeling.

À travers des expériences sur plusieurs jeux de données de télédétection, nous démontrons l'efficacité et analysons les méthodes proposées. La combinaison de ces différents travaux aboutit à un framework robuste et polyvalent pour corriger de manière interactive les cartes de segmentation sémantique produites par des algorithmes d'apprentissage profond en télédétection.

Title : Interactive semantic segmentation of aerial images with deep neural networks
Keywords : Semantic segmentation, Interactivity, Human-in-the-loop, Aerial images

Abstract : We propose in this thesis to build up a collaboration between a deep neural network and a human in the loop to swiftly collect accurate segmentation maps of remote sensing images. In a nutshell, the user iteratively interacts with the network to correct its initially flawed predictions. Concretely, these interactions are annotations representing the semantic labels. Our contributions are fourfold.

First, we propose two interactive learning schemes to integrate user inputs into deep neural networks. The first one concatenates the user annotations with the other inputs of the network (e.g. RGB image). We apply it both to convolutional architectures and to Transformers. The second one uses the annotations as a sparse ground-truth to retrain the network. Then, we propose an active learning strategy to guide the user towards the most relevant areas to annotate. To this purpose, we adapt different state-of-the-art acquisition functions to evaluate the neural network uncertainty. Finally, we propose to modify the algorithm output space to swiftly adapt it to new classes under weak supervision. To alleviate the background shift and the catastrophic forgetting issues inherent to this problem, we compare different regularization terms and leverage a pseudo-label strategy.

Through experiments on multiple remote sensing datasets, we show the effectiveness of the proposed methods and analyze them extensively. Combining these different components results in a robust and versatile framework to interactively correct semantic segmentation maps produced by deep learning algorithms in remote sensing.

Remerciements

Je voudrais tout d'abord remercier mes encadrants : Adrien Chan-Hon-Tong, Nicola Luminari, Bertrand Le Saux et Guy Le Besnerais. Nicola, notamment pour m'avoir accompagné même avant la thèse pendant le stage à Delair puis pour avoir continué tout du long pendant la thèse. Adrien, pour avoir pris l'encadrement en cours de route de façon magistrale. Bertrand, pour avoir cru en moi et en ce projet avant même de m'avoir rencontré, puis en maintenant un encadrement toujours bienveillant depuis l'Italie. Guy, également pour m'avoir fait confiance avant même notre rencontre, et pour l'encadrement, plus distant, mais tout aussi pertinent. Et à vous tous pour nos discussions hors thèse !

Je voudrais ensuite remercier les membres de mon jury qui ont pris de leur temps pour évaluer mon travail. Tout d'abord, Begüm Demir et David Picard, qui ont accepté d'être mes rapporteurs, pour leur lecture attentive du manuscrit et leurs retours pertinents. Ensuite, Céline Hudelot, Charlotte Pelletier et Devis Tuia qui ont accepté d'être examinateurs lors de la soutenance de thèse.

Ensuite, je voudrais remercier l'ensemble des personnes de l'ONERA, permanents (Pierre, Philippe, Frédéric, Aurélien, Élise, Pauline, Martial, Julien, Alexandre, Stéphane, Baptiste), doctorants passés (Rodrigo, Rodolphe, Pierre, Javiera, Guillaume H., Guillaume V.R., Simon, Benjamin, Alexis) et présents (Rémy, Nathan, Laurane, Maxime, Marius, Quentin, Thomas, Pol, Adrien, Philip). Notamment, Philippe pour toutes nos discussions autour de la montagne et pour m'avoir donné envie dernièrement d'aller faire un tour vers la Meije orientale. Pierre F. pour toutes nos discussions et, surtout, pour la peluche d'alpaga ! Pierre G. pour m'avoir appris à jongler ! Rémy, pour avoir été mon co-bureau pendant ces trois années. Maxime et Nathan pour la folle organisation des cagnottes des doctorants précédents et les idées premiums. Élise et Aurélien, merci d'avoir été là ISPS à Nice ainsi que pour ma répétition de la soutenance ! Marius, pour m'avoir sauvé la thèse en étant un intermédiaire précieux entre moi et le Deeplab. Au passage, merci beaucoup à Florence Marie pour m'avoir bien aidé dans toutes les démarches administratives en fin de thèse ! Enfin, Javiera, pour toutes nos discussions sur la thèse, hors thèse et, surtout, pour nos sessions d'escalade à Fontainebleau et ailleurs !

Merci aussi aux doctorants rencontrés lors des conférences qui ont pu se dérouler en présentiel sur la fin de ma thèse ! À Nice, Iris, Mathilde, Romain, Roza et Zoé. C'était sympa les baignades avant les sessions poster ! À Kuala Lumpur, Reza B., Reza A., Sanhita et Amjad. Thanks guys for introducing me to Iranian food !

Je voudrais maintenant remercier Delair/Alteia, déjà pour le financement, mais aussi pour les gens formidables qui en font (ou en ont fait) partie. Yann pour m'avoir pris tout d'abord en tant que stagiaire à Delair puis pour m'avoir aidé à monter ce projet de thèse. Sans toi, rien n'aurait été possible. Toute l'équipe data toulousaine (Nicola (encore !), Hacene, Aurelien, Romain, Rémi, Kaaviya et Moussa) que j'ai vu se former avant et pendant la thèse ansi que l'ancienne équipe data parisienne (Nicolas, Sophie et Théo). Moussa, je te souhaite une excellente thèse ! Aurelien, la dent d'Orlu était exceptionnelle ! Théo, merci de m'avoir ensuite accompagné le premier dans l'univers des sommets, des crevasses et des cabestans : "C'est un relais de montagne, faut pas trop tirer dessus".

Il y a ensuite tous les amis de la fac, du lycée, de la primaire, de la maternelle et d'ailleurs (désolé si j'en oublie) : Paul, Arthur B., Arthur E., Thomas, Alexandre, Franck, Pauline G., Pauline C., Raphael, Marie, Cécile, Babujhi, Louise, Naouzad, Kevin, Jouana, Francois, Guillaume, Emilien, Nicolas, Aude, David, Prédive.

Paul, pour avoir ouvert ton bar en meme temps que ma thèse commencait. Pas de chances, le covid arrivait aussi... Merci Raph d'avoir parcouru avec moi les terres d'Ilyzaelle et merci Alex d'avoir été mon compagnon diablotin.

Et puis bien sûr, merci à tous ceux de l'escalade sans qui cette thèse aurait été peut-être plus studieuse (elle l'était quand même déjà !) mais beaucoup moins heureuse ! Merci donc à Nico, Pepi, Félix, Justine, Charles, Camille, Benoit, Seb, Maxime, Noé, Flo B., Flo J., Marion, Mahdi, Théo (encore !) et Javi (encore !) pour les nombreuses sessions des falaises bretonnes aux Alpes slovènes en passant par les Calanques et, évidemment, par Fontainebleau. D'ailleurs, un petit merci à certains des blocs qui m'ont inspiré et fait rêver pendant ces trois ans : Bleau's Art, La Grande Marche, Noir Désir (gauche), Retour aux Sources et le Gruyère.

Enfin, merci à ma famille et notamment à toi Maman pour m'avoir soutenu et accompagné à tout moment ! Papa, cette thèse t'est dédiée, toi qui est parti juste avant que tout cela ne débute. J'aurais aimé pouvoir partager cette aventure avec toi.

Pour finir, merci à toi Syrielle d'être à mes côtés, que ce soit sous l'eau sur la fin de thèse ou à 6476m au dessus de la mer !

Table of contents

List of Figures

List of Tables

Chapter I - Introduction

Contents

This thesis explores and compares methods to build and enhance synergy between semantic segmentation algorithms and their users. This is a CIFRE PhD, built as a partnership between ONERA, University Paris-Saclay, and Alteia. It is within this context that this research topic was defined, anchoring it into concrete challenges and issues for the company.

I.A . Context

I.A.1 . Artificial intelligence

Computer science, and in particular artificial intelligence (AI), has progressed at an extraordinary rate in recent years. This has allowed the automation of a large number of tasks as well as great advances in robotics, imaging or language processing. For example, we can now automatically translate languages or have driving assistants. AI can even outperform human experts in chess or Go, without explicitly coding the game's rules. However, even without considering the super-powered AI described in some works of science fiction, which can predict the evolution of a civilization (famous novels of I. Asimov [*]) or that endows robots with consciousness, more realistic goals, such as autonomous driving or perfect automatic mapping, have not yet been achieved. Indeed, it is always possible for AI to make mistakes, even on simple tasks. In this context of fallible AI, we seek with this thesis to build a synergy between AI algorithms and human users for the semantic understanding of optical remote sensing images.

I.A.2 . Remote sensing applications

∗. https://en.wikipedia.org/wiki/Psychohistory_(fictional)

Remote sensing data can be used at different scales. Satellite imagery can help to understand our planet at large scale and hopefully prevent climate disasters by monitoring deforestation or glacial melts. Drones act more locally by monitoring for instance industrial infrastructures. More sensational, they can also be used to rescue avalanche victims [†] or to lead lost mountaineers to safety [‡]. Data captured from airplanes or helicopters come between the two and can, for example, monitor the condition of railways or an electrical network on a regional scale. Moreover, remote sensing data comes from a wide variety of sensors. This implies that we are not limited to RGB optical data but that we can also exploit thermal, outside the visible spectrum (e.g. IRRG, UV) or geometric (e.g. DSM from LiDAR acquisitions) data to address even more issues. Machine learning methods are then essential to model and make the most of such data.

I.A.3 . Semantic segmentation

A key issue to scene understanding is semantic modeling, which consists, for example, in detecting objects in the image or in classifying areas of the image. More specifically, semantic segmentation aims at a classification of the image at the pixel level. The stakes are multiple and affect many fields such as autonomous cars or medical imaging. In remote sensing, semantic segmentation applications are numerous : they range from monitoring crop maturation indices or the volumes of stone stockpiles in careers using drones, to the evaluation of natural disasters using satellites, including cartographic mapping at various resolutions.

I.A.4 . Current solution and its limits

Semantic segmentation is currently efficiently addressed with deep neural networks (DNNs) but is still only partially solved. Indeed, even under ideal conditions, state-of-the-art DNNs make errors. Moreover, they are often sensitive to various limitations such as the necessity of training over large, properly annotated datasets or the discrepancies between training and test data. In remote sensing, the latter "domain changes" come from a multitude of reasons : weather conditions, geographical locations, sensor types, etc. As a result, assessing the effectiveness and performances of DNNs is difficult and often makes their industrial deployment complicated. A human intervention may then be necessary to certify and complete their results.

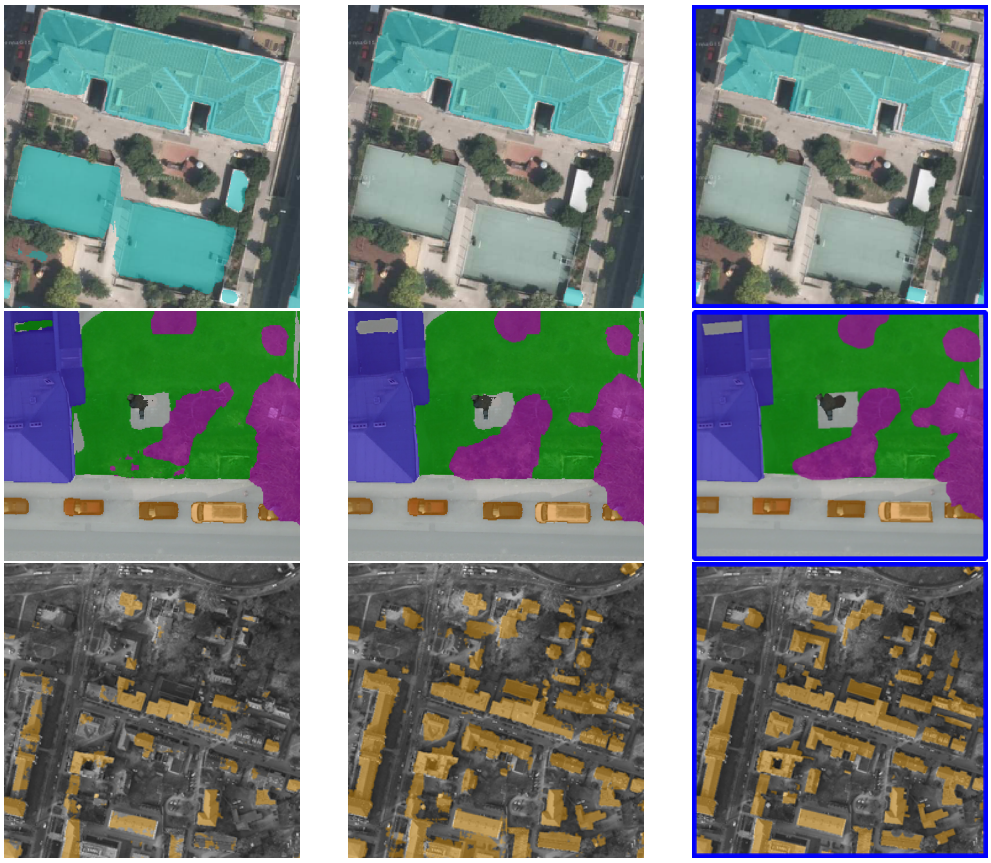
I.A.5 . Proposed solution : Add a human-in-the-loop

To address the limitations of DNNs in semantic segmentation for remote sensing, we deal in this thesis with interactive learning. This paradigm involves a human in the loop working in synergy with a learning algorithm to train it, refine it or adapt it to the user's inputs. Compared to other machine learning algorithms like SVMs,

†. https://www.springwise.com/innovation/sport-fitness/powderbee-drone-avalanche-rescue

‡. https://www.redbull.com/int-en/drone-mountaineering-rescue

DNNs are rather heavy algorithms as they can easily be composed of millions of parameters. It will therefore be necessary to focus on both the fluidity and the performance gain of our methods.



(a) Output segmentation (b) Potentially expected results (c) Ground-truth

Figure 1.1 – Examples of semantic segmentation output by a DNN compared to what could be practically expected for an industrial use-case : it can slightly differ from the ground-truth used for training.

Figure 1.1 shows situations where a network makes partially wrong predictions and we explain here how an interactive scheme could improve these results. We will also provide more examples in Chapter II on Figure 2.12. On the first row, some sport fields are initially wrongly classified as buildings. They could be easily removed from the segmentation map by a user. On the second row, borders of trees, low vegetation and impervious surface are not completely accurate. Precise boundaries of trees are required when the goal is, for example, to compute the distance of trees from infrastructure to determine if they should be pruned or not. In this case, a user could thus refine the border according to its needs. These small changes may have little influence on the evaluation measures but may be necessary

for the task at hand. The third row displays an example of domain adaptation where a DNN has been trained to segment buildings in New-Zealand and is used on a German city. Here, the errors are quite similar (missed buildings and very few false positives) and could therefore be corrected with a few corrections effectively propagated throughout the image. To summarize, in these three examples, the segmentation outputs by a DNN are partially correct but each could be refined by human intervention to match the expected result. Along this manuscript, we will thus explore different ways to let DNNs make the most of user interactions.

I.B . Open research questions

This thesis aims at improving the synergy between the user and semantic segmentation algorithms based on deep neural networks.The objective is to propose methods to learn the semantics of a scene by taking into account the successive annotations of the user while reducing the learning cost (amount of data and learning time). We thus seek to conciliate machine learning in a context of massive data with sparse information provided by the user.

To this end, the following research questions are addressed.

1. **How to interact with neural networks after learning ?**
   In general, neural networks are trained offline and are then used in a fully automatic way. We study how interactivity after training can allow to refine the predictions of these algorithms. Such approaches must imperatively meet two criteria. First, they have to be fast so that interactions with the user are as seamless as possible. It is therefore inconceivable to keep the training data in memory or to re-train the algorithm entirely. Second, they need to efficiently generalize the sparse information provided by the user to the whole scene under study.
2. **How to get relevant data ?**
   User annotations are inherently sparse compared to images, and this is even more true in remote sensing where images can be huge. It can also be particularly tedious for a user to go over each segmentation map in detail. In order to ease the annotator's burden, we study how to guide the annotator through the image by selecting relevant data to annotate.
3. **How to adapt to new data and new use cases ?**

   The standard process in Artificial Intelligence for Earth Observation consists of learning on a given pre-annotated dataset to produce a single static model. However, it is best suited for well-defined and narrow tasks in closed environments. Indeed, while it might work well on benchmarks, it often fails on real-life open and dynamic environments. This is mainly due to the domain shifts inherent to deployment conditions in Earth Observation contexts : different geographical areas, meteorologic conditions, or kinds

of sensors. Moreover, there may also be a need to predict new target classes. Hence, not only the input data distribution may vary but also the label set. These two problems are more generally part of *transfer learning* [Pan and Yang, 2009], which aims to address a specific problem using knowledge extracted from a different but related problem. They are also referred to in the literature as *domain adaptation* [Kellenberger et al., 2021] and *class incremental learning* [Tasar et al., 2019]. They are not trivial because, in these situations, neural networks are then often prone to so-called *catastrophic forgetting* [Kirkpatrick et al., 2017] of previous knowledge and/or to *over-fitting* [Srivastava et al., 2014] on the training data. In addition, we address these problems within our interactive framework.

I.C . Contributions

The main contribution of this thesis is an interactive framework to refine semantic segmentation maps issued by a neural network relying on user annotations. This framework includes the following components :

1. Two algorithms, DISIR and DISCA, to interact with a neural network in the context of Earth Observation, leading to increased performance in a controlled time.
2. An active learning methodology relying on the estimation of neural networks uncertainty to choose the data to annotate.
3. Regularizations and pseudo-labeling mechanisms to apply DISCA to domain adaptation and class-incremental semantic segmentation.

These different components are presented and experimentally validated in the different chapters of this thesis.

What we are not going to do

It should be emphasized that we are not going to seek the best initial DNN for semantic segmentation from pre-annotated data. Indeed, this would require an extensive work beyond the scope of this thesis, notably on the architecture choice and training scheme. Instead, we rather focus on methods agnostic to the architecture choice able to improve, as efficiently as possible, the initial segmentation outputs.

I.D . Manuscript outline

This manuscript consists of seven chapters. Chapter II presents the bibliography study required to clearly position the remainder of this thesis. It also provides the scientific context for this work and introduces the datasets and metrics used in our experiments. Chapter III presents a methodology to interact with convolutional neural networks after training to refine their initial segmentation results. These

modifications do not require any retraining and are spatially localized close to the interactions. Since this spatial limitation is partially due to the convolution receptive fields, we explore in Chapter IV Transformers and self-attention architectures to mitigate it. To further overcome this spatial limitation, we then propose an on-the-fly retraining methodology using the user annotations as a sparse ground-truth. We notably analyze the pros and cons compared to the first methodology and show that it is specifically suited for domain adaptation problems. In Chapter V, we set-up a new methodology relying on active learning to guide the user in its annotation task. We adapt in Chapter VI the retraining methodology from Chapter IV for class-incremental semantic segmentation. To this purpose, we analyze different regularizations and a pseudo-labeling strategy. Finally, we propose an overview of our results and draw perspectives of research in the concluding Chapter.

I.E . Publications

The majority of the contributions presented in this thesis have been published in peer-reviewed articles and communications and their source code have been released publicly. Below is a list of the publications related to this work at the time of writing and the associated code repositories.

Journal articles

— **Lenczner G.**, Chan-Hon-Tong A., Le Saux B., Luminari N. & Le Besnerais G., *DIAL : Deep Interactive and Active Learning for Semantic Segmentation in Remote Sensing*, JSTARS journal https://github.com/alteia-ai/DIAL

Conference & workshop articles

— **Lenczner G.**, Chan-Hon-Tong A., Luminari N. & Le Saux B., *Weakly-supervised continual learning for class-incremental segmentation*, IGARSS 2022 https://github.com/alteia-ai/ICSS

— Chan-Hon-Tong A., **Lenczner G.** & Plyer A., *Demotivate adversarial defense in remote sensing*, IGARSS 2021

— **Lenczner G.**, Chan-Hon-Tong A., Luminari N., Le Saux B. & Le Besnerais G., *Interactive Learning for Semantic Segmentation in Earth Observation*, ECML-PKDD MACLEAN Workshop 2020 *(Best Student Paper Award)* https://github.com/delair-ai/DISCA

— **Lenczner G.**, Le Saux B., Luminari N., Chan-Hon-Tong A. & Le Besnerais G., *Segmentation sémantique d'images aériennes avec améliorations interactives*, RFIAP 2020

— **Lenczner G.**, Le Saux B., Luminari N., Chan-Hon-Tong A. & Le Besnerais G., *DISIR : Deep Image Segmentation with Interactive Refinements*, ISPRS Annals 2020 https://github.com/delair-ai/DISIR

Chapter II - Related work

Contents

Chapter overview

Above all, this thesis is a work of computer vision applied to remote sensing imagery and relies on methods rooted in deep learning with specific interest in the interaction between the human user and the algorithms.

This chapter first provides an introduction to the different general concepts necessary to contextualize globally the contributions of the following chapters. Precisely, we start by giving a brief overview of computer vision and we recall the origin and the components of current convolutional neural networks. Then, we delve

into details of the origin of semantic segmentation and present some of the current neural network architectures made to address it. Finally, we explain the different limits of these architectures and how an interactive approach can address them.

Second, we deal with interactive learning and review work that has inspired our own work or that deals with similar problems. We first present a panel of interactive interpretation works in remote sensing. We then detail both historic algorithms and recent deep learning based algorithms designed for interactive segmentation. We finally describe active learning strategies and specifically focus on uncertainty-based methodologies.

We conclude this chapter with a presentation of the public datasets and metrics we use to evaluate our different works. We also highlight the limitations of deep neural networks in semantic segmentation through examples taken from these datasets.

II.A . Understanding the stakes

II.A.1 . An introduction to computer vision

Computer vision is an interdisciplinary field which aims to teach computers what the human visual system can do. Its origin dates back to the 1960's and scientists thought that solving this problem would be a piece of cake. Indeed, an MIT student was given a summer project to simply connect a camera to a computer and then have him "describe what he saw". As this project was obviously not completed in one summer, computer vision has been an active research topic ever since. There are now many different issues related to computer vision, sometimes crossing boundaries with other research fields. They mostly arise from real world problems :

- Image classification, where the class of an image is selected among different known options ;
- Image captioning, where a textual description of an image is generated ;
- Face or fingerprint recognition ;
- Semantic segmentation, where each pixel of an image is classified among different known options ;
- Point cloud classification/segmentation where the objective is the same as for images but with more complex point cloud data ;
- Visual question answering (VQA) aims to build systems able to answer questions regarding images ;
- Depth estimation on images or point clouds ;
- Object detection in images ;
- Robot localization in unknown environment ;
- Optical flow estimation, in which the movement of each pixel is estimated between different images.

This list is by no means exhaustive but gives a glimpse of the variety of possible

use-cases. As stated in Introduction, we focus here on semantic segmentation and, more precisely, on refining semantic segmentation maps output by deep neural networks. Indeed, these algorithms are now the main algorithms used to address many computer vision tasks.

Although our field of application is remote sensing, our works could most likely be adapted to other fields such as medical imaging. It would also be interesting to adapt our methods to 3D or depth estimation, but this would imply rethinking the format of the user inputs. Finally, similar to VQA, user interaction could come in a textual form.

II.A.2 . From perceptron to convolutional neural networks

II.A.2.a History : perceptron and back-propagation

Loosely inspired by neuroscience, deep neural networks have their origin in the Rosenblatt Perceptron [Rosenblatt, 1958]. This algorithm is built around the artificial neuron concept. This neuron is a function f that, given its weights $\mathbf{w} \in \mathbb{R}^N$ and a bias $b \in \mathbb{R}$, maps its input $\mathbf{x} \in \mathbb{R}^N$ to an output $y \in \mathbb{R}$:

$$f(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{w} \cdot \mathbf{x} + b > 0, \\ -1 & \text{otherwise} \end{cases}$$

Originally composed of a single neuron, this Perceptron is therefore called single-layer perceptron and can be trained with the Perceptron Learning Rule :

$$W_i' = W_i + \lambda(Y_t - Y)X_i$$

where W_i' is the i updated weight, W_i the i current weight, λ the learning rate, Y_t the expected output, Y the current output and X_i the input i. This algorithm can't successfully be applied to several neurons in a row and is thus not able to solve complex non-linear problems. To address this problem, the back-propagation algorithm [Rumelhart et al., 1986] computes the gradient of the loss function with respect to each weight of the network using the chain rule which expresses the derivative of the composition of two differentiable functions in terms of each of their derivatives. Combined with an optimizer such as Stochastic Gradient Descent (SGD) [Robbins and Monro, 1951], it enables to efficiently train deep neural networks such as multi layer perceptron (MLP) and convolutional neural networks (CNNs) to solve complex practical cases. It remains, to this day, the standard way to train deep neural networks.

Non-linear activation functions are another key ingredients in deep neural networks. Indeed, without requiring any additional parameters, they follow neuron layers to add non-linearity in the networks, which enables the learning of complex patterns. Common activation functions include sigmoid function, hyperbolic tangent or rectified linear unit (ReLU).

II.A.2.b Convolutional neural networks

CNNs are artificial networks particularly relevant when applied on visual data since their underlying principle cleverly handles the image structure. Indeed, contrary to MLPs in which all neurons are connected to each other, CNNs are based on the shared-weight convolution kernels that slide along input features to output feature maps. They also usually rely on pooling layers to reduce the size of the processed images by summarizing the information locally. This leads to a reduction of the number of the parameters and of the computation cost.

However, due to their large amount of parameters, CNNs simply composed of convolution, activation and pooling layers are prone to many convergence issues. Indeed, they can be very sensitive to a poor initialization of the weights and are particularly inclined to over-fitting. Moreover, in large networks trained with back-propagation, gradients can get extremely small, partially due to the activation functions. This prevents the weights to update their values and thus the convergence of the algorithm. This phenomenon is known as the vanishing gradient problem.

Several landmark works have led to algorithmic elements that are now considered essential for the convergence of CNNs. Batch normalization [Ioffe and Szegedy, 2015] layers re-center and re-scale the layers inputs to protect networks against convergence issues related to their initialization. Dropout layers [Srivastava et al., 2014] randomly drop neurons and their connections during training to prevent neurons to co-adapt too much. This acts as regularization and is an effective tool to reduce the over-fitting problem. Introduced by the popular ResNet architecture [He et al., 2016], a residual connection directly pass gradient information from a layer to a deeper layer in the network. By enforcing a better gradient flow, this prevents the vanishing gradient problem and enables to train large neural networks.

To summarize, current CNN architectures are usually composed of :

— Convolution layers to extract features from their input ;
— Activation layers (e.g. ReLU) to add non-linearity in the network and to decide which neurons should be activated ;
— Pooling layers to reduce the size of the processed image and thus the number of parameters and computation cost ;
— Dropout layers to prevent over-fitting ;
— Batch normalization layers to stabilize CNNs convergence ;
— Residual connections to better propagate information across the different layers.

CNNs achieve their first success when Yann LeCun *et al.* [LeCun et al., 1989] proposed to apply them to automatically read handwritten digits. However, partially due to a lack of large annotated datasets and of computer power, they only truly earn their popularity in 2012 when [Krizhevsky et al., 2012] win the ImageNet competition with the AlexNet architecture. Alongside with the advent of powerful GPUs and large datasets like ImageNet or Pascal VOC [Everingham et al., 2010], this really throws CNNs into the spotlight. As already mentioned, they are now widely

used to solve most computer vision problems, including semantic segmentation.

II.A.3 . Semantic segmentation

(a) Input image (b) Semantic segmentation (c) Instance segmentation

Figure 2.1 – Comparison between image, semantic and instance segmentation on rock segmentation from Fontainebleau forest.

II.A.3.a Origin of semantic segmentation

As previously defined, semantic segmentation simply consists in a pixel-wise classification of the image. Even though the name is quite new, coined in the 2010s, the concept of pixel-wise classification is much older.

In remote sensing literature, it is often referred to as image classification and researchers have been working in this field since the beginning of the 1990s [Civco, 1993, Inoue et al., 1993]. It now must not be confused with the global image classification concept popularized by ImageNet [Deng et al., 2009]. Indeed, the classification of whole natural images makes more sense than for remote sensing images which generally contain many objects.

The first public datasets annotated for pixel-wise image classification in remote sensing also date from this period. For instance, the Indian Pines dataset from 1992 [Baumgardner et al., 2015] consists of a single image of 145×145 pixels with a resolution of 20m/pixel. In comparison and as described in Section II.C, remote sensing datasets can now easily contain billions of annotated pixels at resolutions below 50cm.

In computer vision, semantic segmentation was first approached in a class-agnostic form, usually called image segmentation. In this case, the objective is to group the pixels into meaningful sets that belong to the same objects, without any classification. Semantic segmentation adds, as the name implies, a semantic that describes these groups of pixels. To go even further, instance segmentation also identifies the unique occurrences of objects of different categories. This adds a lot of complexity to the problem, especially when the objects are close or overlapping, and we do not address this problem in this thesis. Figure 2.1 depicts the difference between these different segmentation levels.

Let just point out that pixel-wise classification has also been extended to super-pixel classification to group pixels into meaningful regions using ad hoc clustering

methods [Achanta et al., 2012] but this literature will not be reviewed here.

II.A.3.b Pioneering segmentation methods

Like other computer vision tasks, image segmentation is first addressed using algorithms relying on hand-crafted rules and features and we present here some works that have been real milestones. In 1991, a break-through is achieved by [Vincent and Soille, 1991] who first apply the watershed method on image segmentation, regarding images as a topographic landscape with ridges and valleys defined by the gradient magnitude of the pixels. In 2000, [Shi and Malik, 2000] propose to treat image segmentation as a graph partitioning problem and proposed to segment the graph with the normalized cut criterion, which measures both the total dissimilarity between the different groups as well as the total similarity within the groups. Meanwhile, the development of data-driven methods for semantic segmentation emerged. In these early works [Roli and Fumera, 2001], pixels are considered as intrinsically distinct elements. These pixels are thus classified without any context or semantic awareness using classification algorithms. In remote sensing, these methods are suitable for small datasets with coarse pixel resolution. Indeed, pixels are less correlated to their neighbors than at high resolution and there is a lack of training data to really leverage contextualization benefits. However, for the current larger and highly resolute datasets, it is important to be able to contextualize the different pixels to reach accurate classification.

The methods [Shotton et al., 2009] that followed usually rely on local descriptors like SIFT [Lowe, 2004] descriptors. These allow a better contextualization and thus a better classification of the pixels, but they are hand-crafted and they still do not reach the capabilities of the learned representations of deep neural networks.

II.A.3.c DNNs

CNNs are first applied to semantic segmentation by predicting the label of each pixel using a sliding window centered on it [Ciresan et al., 2012]. This patch-based approach is inherently computationally heavy. To address this issue, [Long et al., 2015] propose the Fully Convolutional Network (FCN), which is the first end-to-end neural network to directly process the whole image for semantic segmentation. As depicted in Figure 2.2, it encodes the image into a latent space of reduced spatial dimension and then performs a transposed convolution to recover the initial resolution, producing a rather coarse segmentation map. To address the coarse decoding issue, U-Net [Ronneberger et al., 2015] introduces a U-shaped architecture composed of an encoder and a symmetrical decoder, as shown in Figure 2.3. The encoder first gradually encodes the image and reduces its spatial dimension with pooling layers. Then, the decoder gradually recovers the object details and spatial dimension. This architecture also has connections between the encoder and the decoder layers, enabling a better information propagation. This architecture is

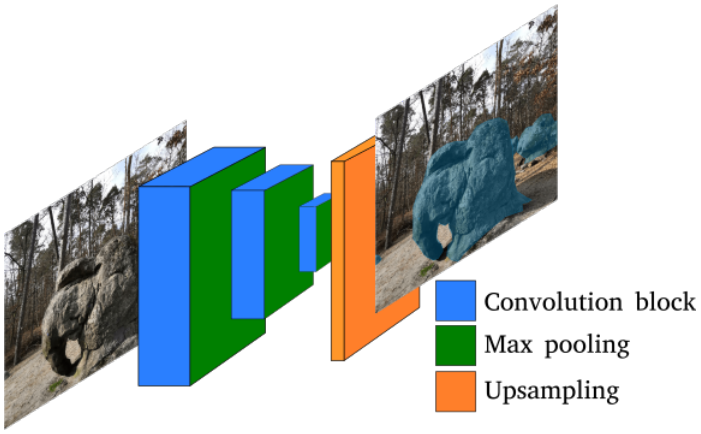


Figure 2.2 – Fully Convolutional Network (FCN) architecture proposed by [Long et al., 2015].

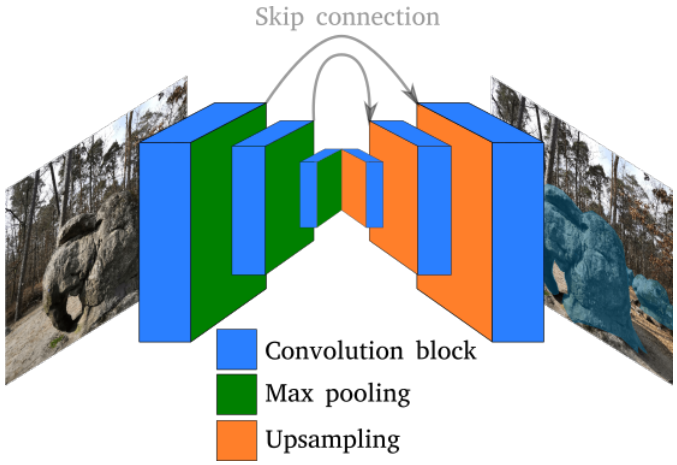


Figure 2.3 – U-Net architecture.

still widely used today, with more powerful encoders and decoders than the originals. We mostly use the LinkNet [Chaurasia and Culurciello, 2017] architecture in this thesis, which is a slight variation of U-Net using ResNet [He et al., 2016] encoder. To alleviate the decoder computation burden, SegNet [Badrinarayanan et al., 2017] stores the pooling indices in the encoder to freely decode the encoded segmentation map using unpooling layers. Its architecture is represented on Figure 2.4.

To name a few of the many works that have applied and adapted these architectures to remote sensing problems, [Kampffmeyer et al., 2016] focus on the segmentation of small objects and thus on class imbalance for accurate land cover mapping. [Audebert et al., 2016] propose a data fusion strategy to jointly use optical and laser data and [Kemker et al., 2018] propose an adaptation of these architectures to multi-spectral imagery. For a more comprehensive overview of semantic segmentation works in remote sensing, we refer the interested reader to

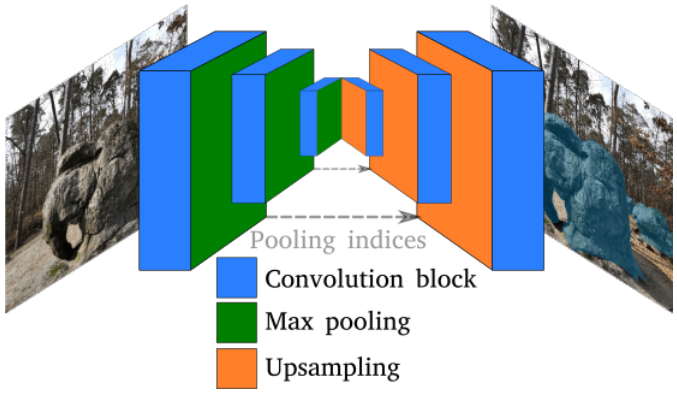


Figure 2.4 – SegNet architecture.

this survey [Yuan et al., 2021].

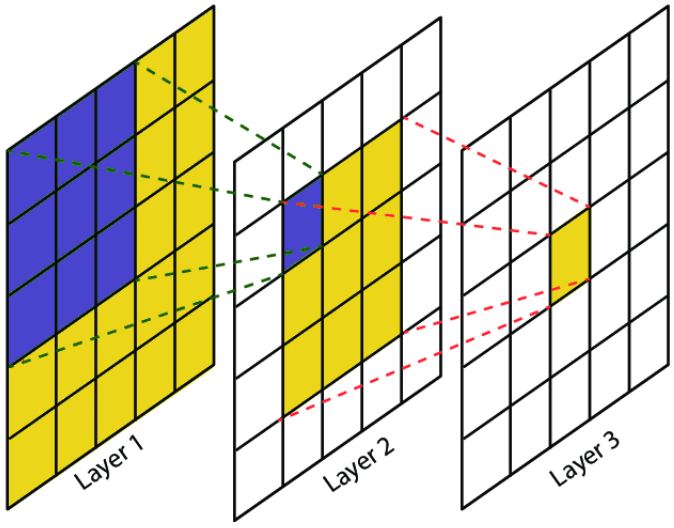
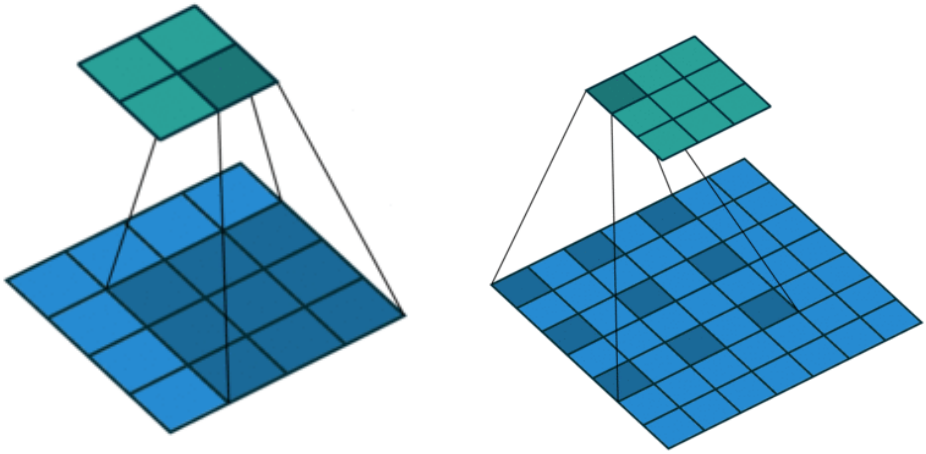


Figure 2.5 – The receptive field of each convolution layer with a 3x3 kernel. The purple area marks the receptive field of one pixel in Layer 2, and the yellow area marks the receptive field of one pixel in Layer 3. Reproduced from [Lin et al., 2017].

A significant limitation of the convolutional neural networks is the spatial limit of their receptive field : for a given pixel, it is the maximum distance in which the other pixels are involved in the classification. This phenomenon is sketched in Figure 2.5. This defines the taking into account of the context of each pixel for their classification. The global receptive field of an architecture is defined by the receptive field of its convolutional layers. Each of them is originally defined by their kernel size but increasing it is not appealing, since the number of parameters scales with the square of the kernel size. [Yu and Koltun, 2016] propose to use atrous convolution (or dilated convolutions, described in Figure 2.6) to make the global receptive field of the network grow exponentially with the number of layers instead



(a) Classic convolution (3x3 kernel) (b) Dilated convolution (3x3 kernel, dilation rate 2)

Figure 2.6 – A dilated convolution enables a larger receptive field than a convolution with the same amount of parameters. Reproduced from [Dumoulin and Visin, 2016].

of linearly. DeepLabv3+ [Chen et al., 2018] uses these in conjunction with a spatial pyramid pooling module [He et al., 2015] that pools the input at different spatial resolutions for better context encoding. Illustrated in Figure 2.7, this architecture is still widely used today thanks to its high context capture ability.

In addition to the receptive field of the networks, another problem for taking into account the context is the size of the images to be processed in remote sensing. Indeed, remote sensing images are often too large to be processed directly. They are therefore generally tiled to be processed by patches. Thus, even without considering the receptive field limitation, the taking into account of the context will be constrained by the size of the patches, which is directly linked to the available computing power.

We will now discuss the other main limitations of DNNs and detail some existing paradigms to address them.

II.A.4 . First limitation of DNNs : A lack of training data

II.A.4.a Reasons for this lack

As mentioned in Introduction, deep neural networks still make mistakes despite their complex and powerful architectures. This is often partly due to **a lack of well annotated training data** :

— It is extremely costly to annotate semantic segmentation data since each pixel has to be annotated. For instance, the price to annotate an image for semantic segmentation is approximately 50 times higher than for image classification with Amazon Mechanical Turk [*]. Besides, since remote sensing

*. https ://aws.amazon.com/sagemaker/data-labeling/pricing/

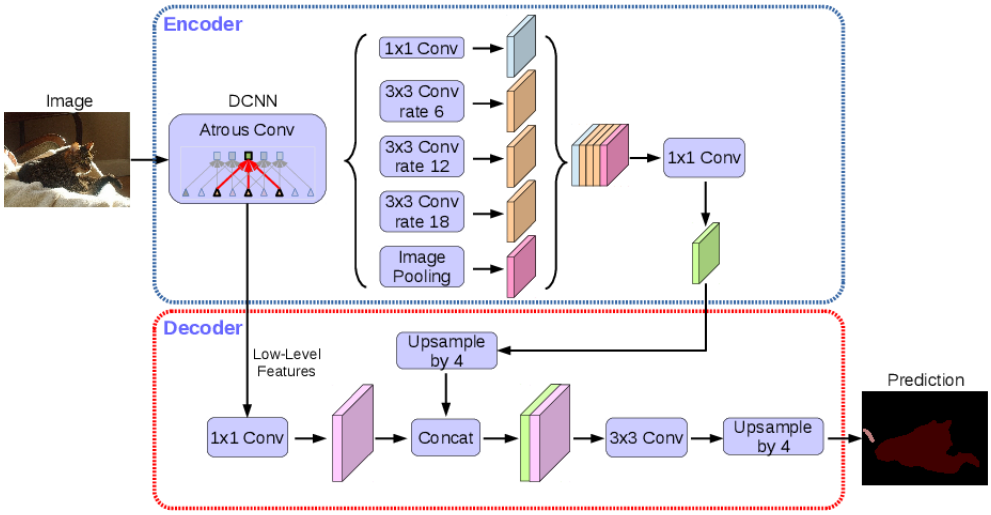


Figure 2.7 – DeepLabv3+ architecture. Reproduced from [Chen et al., 2018].

images are generally larger than natural images, the cost of annotation is even higher.

— It is also laborious to correctly annotate the borders of the different objects, especially since annotations can be subjective and depend on the annotator. For instance, should a pant segmentation include the belt or not ?

— Solutions for automatic annotations sometimes exist, for example by using cadastral data for building annotation. However, although allowing to save considerable time, the resulting annotations are often inaccurate and lead to partially erroneous ground truth maps.

For these reasons, correctly annotating a dataset for semantic segmentation requires considerable rigor and time. For instance, the CityScape dataset [Cordts et al., 2016] required more than 1.5 hour on average per image to annotate them precisely. We will see that our methods presented in Chapter III and Chapter IV swiftly lead to accurate results in a semi-automatic way.

II.A.4.b Semi-supervised learning

This inherent cost of annotating data for semantic segmentation often results in datasets with fewer labels than data. In this scenario, called **semi-supervised learning**, training typically boils down to learning the most of the available labels while leveraging unlabeled data to learn a better inner representation as support. This can be done trough auxiliary tasks such as predicting rotations [Gidaris et al., 2018], solving the Jigsaw puzzle [Noroozi and Favaro, 2016] or inpainting [Pathak et al., 2016]. The data distribution can also be modeled using generative models like GANs [Souly et al., 2017] or energy-based models [Castillo-Navarro et al., 2021b]. Consistency regularizations, which enforce an invariance of the neural network predictions over different perturbations applied to the in-

put images, usually combined with pseudo labeling strategies, have been applied with good results (e.g. FixMatch [Sohn et al., 2020], CutMix-Seg [French et al., 2020], CCT [Ouali et al., 2020], MeanTeachers [Tarvainen and Valpola, 2017]). Drawing inspiration from these works could improve the methodology we propose in Chapter Chapter VI with strong and consistent pseudo-labels. Finally, contrastive learning approaches [Alonso et al., 2021, Chen et al., 2020] aim to learn such an embedding space in which similar sample pairs stay close to each other while dissimilar ones are far apart.

II.A.4.c Weakly-supervised learning

It is also possible to only have so-called **weak labels** available, i.e. labels that do not fully characterize the data for the objective task. In semantic segmentation, weak labels can take different forms like incomplete ground-truth maps, bounding boxes, scribbles, points or image level labels. Semantic segmentation with point supervision was first proposed by What's the Point [Bearman et al., 2016] (WTP) which trains a model from scratch using cross entropy loss on the point labels. Recently, [Hua et al., 2021] propose a regularization relying on neighborhood structures for point-supervised semantic segmentation in remote sensing while [Li et al., 2021] propose to train a network with image-level labels and generated pseudo-labels. In an interactive context, user input often comes in the form of a weak label for a smooth process and we will see, that, inspired by WTP, we mostly rely on point annotations in our different works.

II.A.5 . Second limitation : Transfer learning

The transfer of knowledge from one use case to another is a common machine learning issue and it has been an active research field for a long time. Several works [Pan and Yang, 2009, Weiss et al., 2016, Zhuang et al., 2020] have thus surveyed this field over the years, and some with a particular focus on DNNs [Tan et al., 2018]. These algorithms are indeed particularly sensitive to this problem, especially since they are also prone to the **catastrophic forgetting** of previously learned knowledge and to the **over-fitting** of the training data. As we have previously explained, annotating data for semantic segmentation is expensive, so the ability to efficiently transfer knowledge from DNNs would be very beneficial. In this manuscript, we will address two aspects of transfer learning in semantic segmentation : the adaptation of a model to the distribution shift between the training and test data and the addition of a segmentation class after the beginning of the training.

II.A.5.a Unsupervised domain adaptation

The **distribution shift between training and test data** is often the source of DNNs's mistakes. The goal of unsupervised domain adaptation is thus to adapt a model from a source domain to a target domain with no available labels on the

target domain. In remote sensing, the shift can be due to weather, sunlight, seasons, geographical locations or sensor types.

Different datasets proposed by the community highlight this problem and allow researchers to efficiently build methodologies and algorithms to address it. The ISPRS datasets [Rottensteiner et al., 2012] include images from Potsdam and Vaihingen, which are both in Germany and share the same classes. However, Potsdam images are RGB while Vaihingen's images are IRRG, their resolution are different and the Potsdam acquisition has been conducted in winter so there are no leaves on the trees. For these reasons, the transfer between the two cities is tedious. The INRIA Aerial Image Labelling dataset [Maggiori et al., 2017] proposes building segmentation between cities scattered around the world to directly address the transfer issue. The MiniFrance dataset [Castillo-Navarro et al., 2021a] allows to study the transfer between different areas of France while adding the problem of the lack of labels on some parts of the dataset. Finally, the LandCover dataset [Boguszewski et al., 2021] offers multi-class segmentation on high resolution images from all over Poland.

To address these complex problems, [Kellenberger et al., 2021] provide an overview on domain adaptation in remote sensing and divides the existing methods into three main categories :

— Methods that adapt the latent representation at a given layer of the network between two distributions. For instance, [Barbato et al., 2021] enforce constraints based on clustering and norm alignment on the feature vectors corresponding to source and target samples.

— Methods that adapt the inputs to align source and target distributions, often relying on the generation of adversarial examples [Tasar et al., 2020b] or on the adaptation of the image statistics of the target domain [Hoffman et al., 2018].

— Methods that use a few selected labels from the target domain, usually relying on active learning [Settles, 2009] for the selection of new labels. We will detail active learning in Section II.B.3. The methods that we propose in Chapter IV and Chapter V fall into this category.

Recently, [Lucas et al., 2021] addresses semi-supervised domain adaptation, where some labels are available in the target domain. They notably design a custom regularization weighted by the proportion of labeled data to address domain adaptation with a partial lack of labels. This regularization strategy is similar to the one we propose in Chapter IV.

II.A.5.b Class-incremental semantic segmentation

Class incremental semantic segmentation intends to modify the output space of learning algorithms to add new label classes. This problem was already considered two decades ago [Bruzzone and Prieto, 1999]. In deep learning, the structure of the neural networks makes it often necessary to consider a new network to learn the new classes [Tasar et al., 2019], since it is necessary to change the last layer to

modify the output space. There are then two main identified pitfalls.

First, as in other class-incremental tasks, it is important to prevent the catastrophic forgetting of previously acquired knowledge. Second, specific to semantic segmentation, the new class usually comes from the background, which causes a discrepancy with the previously learned background class. This phenomenon is referred as background-shift and was first identified and tackled by [Cermelli et al., 2020].

Earlier methods used to address these two problems with DNNs by simply storing previous examples [Tasar et al., 2019] but more recent works consider this constraint to be too restrictive due to limited storage or for security reasons. Hence, more recent works design customized regularizations to address the two identified pitfalls. [Douillard et al., 2020] propose distillation losses which enforce statistical matches between the networks. By definition, a distillation loss intends to transfer information between two networks. On the other hand, inspired by few-shots and contrastive learning, [Michieli and Zanuttigh, 2021] rely on prototypes representing the different semantic classes to distinguish them smoothly.

In Chapter VI, we draw inspiration from these different works to interactively add a possible segmentation class. In our framework, we will also focus more on the background-shift issue than on catastrophic forgetting.

The combination of class incremental and domain adaptation gets close to real and difficult use cases in which both changes of objectives and of data distribution may occur. This problem, also called open-set learning, has been little discussed in the literature due to its complexity. Indeed, it requires very flexible models in order to address the combined problems of class-incremental and domain adaptation. It was introduced by [Panareda Busto and Gall, 2017] and it has been recently addressed in several remote sensing works [Tasar et al., 2020a, Dang et al., 2019, Al Rahhal et al., 2022]. However, although this is our ultimate goal, it is necessary to go step by step ; this is why we focus on its sub-problems combined with interactivity.

Now that we have presented the classical semantic segmentation algorithms we rely on and their limitations, we consider the addition of a human in the loop to interactively correct the segmentation maps proposed by neural networks, focusing on interactive segmentation and active learning which inspired our works.

II.B . Interactive learning

II.B.1 . Interactive interpretation in remote sensing

Interactive interpretation of remote sensing data has a long history, partially due to the lack of reference data for training in that field. Interactivity has been processed by various techniques to enhance data mining tools with relevance feedback capability : Bayesian modeling of sample distributions is at the core of VisiMine [Aksoy et al., 2004], a system for data mining and statistical analysis of remote sensing images. Support Vector Machines (SVMs) are used in

[Ferecatu and Boujemaa, 2007] for interactive satellite image retrieval. More recently, boosting has been the method of choice due to the possibility to train quickly in an incremental manner. [dos Santos et al., 2013] use it in binary segmentation to better separate the two classes based on user feedback. [Le Saux, 2014] propose to interactively design object detectors, notably relying on gradient boosting to address the unbalanced and partially mislabeled training data inherent to interactivity.

Active learning, or in other words looking for examples which are the more able to lead to a better classification, is also used for smoother training [Demir et al., 2010, Bruzzone and Persello, 2009]. [Tuia et al., 2011b] survey three main families of active learning strategies for image classification in remote sensing using SVMs : committee, large margin and posterior probability-based. While the large margin family is specific to SVMs, the committee family, which consists of quantifying uncertainty based on the variance between different classifiers, and the posterior probability-based family are the core of current active learning strategies. We delve into further details of active learning in Section II.B.3.

II.B.2 . Interactive segmentation

As shown on Figure 2.8, this problem consists in extracting the foreground of an image with minimal user interaction. The user inputs thus allow refinements according to a predefined behavior.

II.B.2.a Standard tools

(a) Input image with user annotations. (b) Output mask.

Figure 2.8 – Interactive segmentation using GrabCut [Rother et al., 2004]. An initial bounding box is provided to delineate the target object and foreground (green) and background (red) annotations refine the resulting mask.

Relying on a graph partitioning of the images like [Shi and Malik, 2000], GraphCut [Boykov and Jolly, 2001] proposes the first algorithm for interactive segmentation. Improving this approach, GrabCut [Rother et al., 2004] estimates the color distributions of the foreground and of the background with a Gaussian mixture

model to construct a Markov random field over the pixel labels. It then runs a GraphCut based optimization to infer their values. However, this approach has several limitations. Indeed, it is not extendable to multi-class segmentation, is not suited to complex remote sensing images which usually contain many areas of interest and it requires to relearn the model for each new image. However, as illustrated on Figure 2.8, it can provide satisfying image segmentation results without requiring an initial training set. Relying on random forests, [Saffari et al., 2009] use Haar-features for interactive segmentation and [Yao et al., 2012] use Hough-features for interactive object selection. These algorithms come with the speed and direct extension to multi-class classification of random forests but they lack the great ability to characterize images specific to neural networks.

II.B.2.b Modern tools for natural images

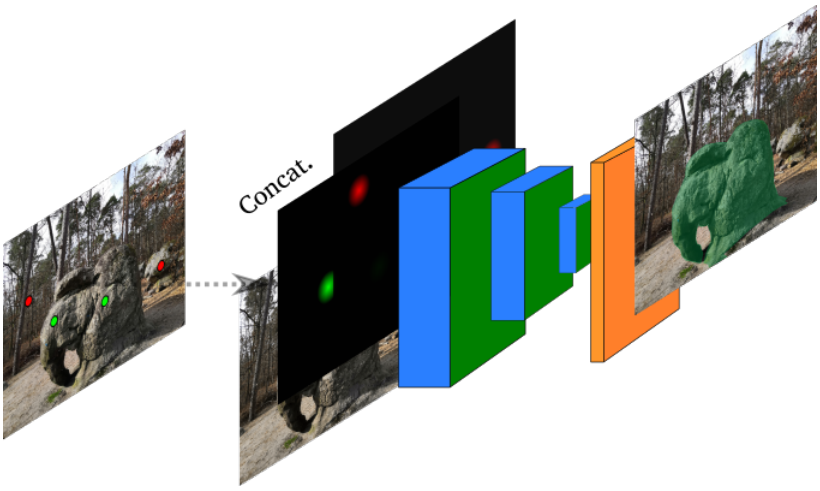


Figure 2.9 – Visual representation of DIOS. User annotations (foreground in green, background in red) are concatenated with the input image to lead to foreground segmentation.

Deep Interactive Object Selection (DIOS) [Xu et al., 2016] is the first proposal of an interactive segmentation framework based on neural networks. It aims for binary classification. In a nutshell, the network takes as input two additional channels concatenated with the RGB image. The first one contains annotation points from the foreground while the other one contains background points. These annotation points are encoded into euclidean distance maps. The annotations are automatically sampled during training using the ground-truth maps. This algorithm is represented on Figure 2.9. Multiple existing works are inspired by DIOS.

A first challenge is the various scales of the target objects. [Liew et al., 2017] adopt a multi-scale strategy which refines the global prediction by combining it with local patch-based classification. [Hu et al., 2019] also follow a multi-scale strategy

by designing a two-stream fusion network to process the annotations differently than the image.Taking it a step further, [Liew et al., 2019] propose a set of scale-varying segmentations to let a user easily choose between a part of an object, the whole object or a group of objects. [Liew et al., 2021] focus on the segmentation of thin objects like bicycle spokes by adding a stream in the neural network to better capture fine-grained details.

A second challenge is to get enough useful annotations. For this purpose, [Mahadevan et al., 2018] use a hard-sample mining strategy at training by selecting annotations among erroneous predictions. Another possible strategy is to maximize the information coming from the different clicks to need a minimum of interactions. Hence, [Lin et al., 2020] enforce special attention on the first click. [Zhang et al., 2020] use only three clicks to segment an object : 2 points for the bounding box encompassing the object and a central point representing it. [Chen et al., 2021b] leverage features and color similarity to better propagate the labeled information of clicks.

Alternatively, [Jang and Kim, 2019] iteratively optimize the annotation maps given as inputs by back-propagating the errors between predictions and annotations. Building on this idea, [Sofiiuk et al., 2020] optimize auxiliary variables instead of the network inputs. This allows to only run forward and backward propagation on a small subset of the network instead of on the entire network. [Kontogianni et al., 2020] use the annotations as sparse ground truth maps to interactively adapt the whole neural network to a specific object.

Then, some approaches take slightly different inputs. Instead of using clicks inside the objects, DEXTR [Maninis et al., 2018] and [Wang et al., 2019b] both ask the user to click points on the borders and corners of the objects. Besides user's clicks, [Ding et al., 2020] leverage phrase expressions as an additional input to better infer the target objects.

Finally, [Benenson et al., 2019] assess DIOS efficiency in the first large scale study of interactive instance segmentation with human annotators. Their experiments hint that center annotation clicks are the most robust and that distance transform to encode the annotation points can be replaced by binary disks.

Our work on the interactive refinement of segmentation maps presented in Chapters Chapter III and Chapter IV is directly inspired by the research presented above. Differently from these works, we rely on interactions to refine existing segmentation maps rather than creating a segmentation from them.

Polygon-RNN++ [Acuna et al., 2018] is an interesting alternative to DIOS-like approaches. Using a CNN-RNN architecture, they predict a polygon which can be refined by moving its vertices. Using Graph Convolutional Networks (GCN), Curve-GCN [Ling et al., 2019] extends this work by predicting a spline which better outlines curved objects. Note that these aforementioned approaches aim to binary classification.

Multi-class interactive semantic segmentation has also been approached in va-

rious ways. Several older methods [Nieuwenhuis et al., 2014, Nieuwenhuis and Cremers, 2012] address this problem using a bayesian maximum a posteriori (MAP) approach while [Santner et al., 2010] rely on a random forest classifier. Recently, [Andriluka et al., 2018] use a combination of two slightly modified Mask-RCNN [He et al., 2017] to compute multiple fixed segmentation propositions and then let the user choose which of these propositions should form the final segmentation. Finally, [Agustsson et al., 2019] are the first to propose a deep learning approach which lets the user correct the shape of a proposed multi-class segmentation. Their algorithm takes as input a concatenation of the image and the extreme points of each instance in the scene and then corrects the segmentation proposal using scribbles.

II.B.3 . Active Learning

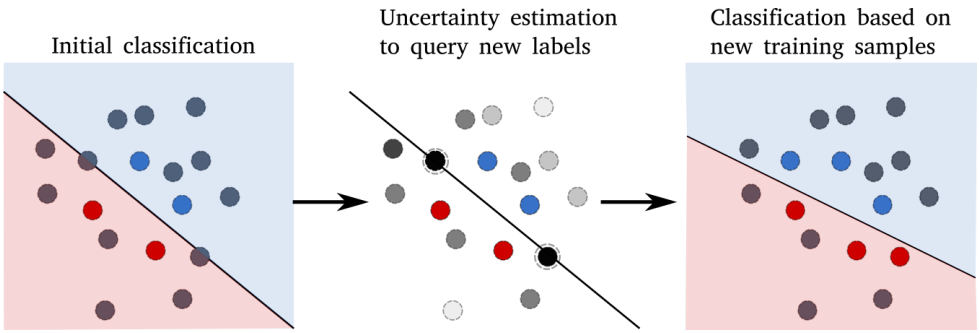


Figure 2.10 – Visual representation of an active learning algorithm. In the middle, the unlabeled samples are ranked according to a specific heuristic represented by the shades of grey. On the right, the selected samples are labeled and added to the training set.

Active Learning aims at optimizing the training process of a learning algorithm through an iterative collaboration with a human oracle. Thus, the user intervenes during the learning process and not afterwards, as in the methods mentioned above. [Settles, 2009] present an exhaustive review of active learning strategies before the deep learning era. In a nutshell, an active learning strategy makes the algorithm choose from a pool of unlabeled data which ones would be the most relevant to improve itself. Then, the oracle provides the asked labels and the algorithm can learn from it. As it defines how to select the data samples to annotate, the acquisition function is the key differentiating component of these methods. These acquisition functions usually rely either on an uncertainty or a representativeness score computed directly with the model to select the most relevant samples. Sketched in Figure 2.10, uncertainty strategies can rely on different criteria like entropy [Shannon, 1948] or disagreement between ensemble models [Hansen and Salamon, 1990] to estimate the model's prediction confidence. As uncertainty-based methods do not aim to be representative of the dataset, they can select very similar examples. To address this issue, representativeness-based methods aim to select the samples in order to form a subset as representative as possible of the entire dataset. Addressing this as

a core-set approach, [Sener and Savarese, 2018] solve it like the K-center problem using the L2 distance between the activations of the final fully-connected layer of the CNN.

Aiming to easily spot wrong predictions, we focus on uncertainty measures in Chapter V to optimally guide the agent toward relevant areas to annotate.

In the past decade, active learning has been deeply explored in remote sensing to train algorithms for animal detection [Laroze et al., 2018, Kellenberger et al., 2019], image classification [Demir et al., 2010, Bruzzone and Persello, 2009] and recently for change detection [Ružička et al., 2020]. [Tuia et al., 2011a] also use active learning to specifically address distribution shift between training and test remote sensing data. [Demir and Bruzzone, 2014] propose an active learning strategy for remote sensing image retrieval using SVMs with an acquisition function combining uncertainty, diversity and density of images in the archive. Finally, ALCD [Baetens et al., 2019] applies a supervised active learning strategy for cloud detection in satellite data which lets the user select the new training samples to annotate.

II.B.3.a Confidence level for active learning

Uncertainty quantification, or confidence estimation, is a long-standing problem in machine learning and has many applications such as out-of-distribution (OoD) sample detection [Liang et al., 2018], the decision to trust the model or to defer to a human expertise in fields like healthcare or the detection of new classes in class-incremental learning [Rebuffi et al., 2017]. Notably, it can also be used in active learning to determine which samples should be sent to the oracle for annotation. Many methods to estimate the uncertainty in deep neural networks have been recently proposed, and they often fall into one of these four categories.

Softmax probabilities. The first category of methods uses the probabilities from the softmax output space of the neural networks. Indeed, [Hendrycks and Gimpel, 2017] propose a simple yet strong baseline using the maximum class probability as an uncertainty estimation and apply to outliers detection. However, it is now well-established that softmax probabilities are prone to different issues such as poor calibration [Guo et al., 2017] and not fit to differentiate in- from out-of-distribution samples [Hendrycks and Gimpel, 2017]. To overcome these issues, [Liang et al., 2018] propose ODIN to detect outliers with a tempered softmax and with adversarial inputs to better distinguish inliers from outliers. Similarly, [Lee et al., 2018] perturb their inputs but instead uses the representation space before the softmax layer and the Mahalanobis distance to do the split.

Model ensembling. Due to its intuitive concept and ease of implementation, another popular class of methods estimates the confidence associated to a sample by measuring the disagreement of different models. This model ensembling can either be explicit and use different models [Beluch et al., 2018] or implicit to be less computationally greedy with one stochastic model using dropout

[Gal and Ghahramani, 2016] (MC Dropout) or batch normalization [Ružička et al., 2020]. However, all these methods inherently require several forward propagation and are thus relatively slow, making them not engaging for interactive interpretation.

Auxiliary models. Other recent approaches design an auxiliary model to learn the uncertainty of the downstream model. These methods do not require to retrain the downstream network and can thus be easily plugged into any pre-trained architecture. While [DeVries and Taylor, 2018] mostly focus on OoD detection, [Corbière et al., 2019] address failure prediction and propose ConfidNet, a neural network to predict if the prediction from the downstream network is accurate or not. However, they are computationally heavy and require a new training phase for each new task and model. In remote sensing, [García Rodríguez et al., 2020] successfully apply the ConfidNet method for land cover segmentation. Instead of directly predicting the downstream network confidence using ground truth, [Besnier et al., 2021] propose to train a second network using the Kullback-Leibler dissimilarity between the outputs of the two networks for confidence estimation. The training of the second network can be done in a supervised or self-supervised manner, which can be extremely helpful depending on the amount of annotated data.

Customized loss. Finally, some works design a specific loss to learn the uncertainty directly during training. For instance, [Yoo and Kweon, 2019] train a model to predict the loss associated to a prediction and [Moon et al., 2020] propose a loss which regularizes the class probabilities to better estimate uncertainty. These methods are computationally efficient and model agnostic but require a full training from scratch and can't be plugged in a pre-trained model.

II.C . Metrics & datasets

II.C.1 . Metrics

In order to measure the actual contribution and the validity of any learning algorithm, it is necessary to properly evaluate its performance at its learning task. Several evaluation metrics exist in the field of semantic segmentation and they are usually variations of pixel accuracy (PA) or of the Intersection over Union (IoU). Details on variations of these semantic segmentation metrics are provided in this survey [Garcia-Garcia et al., 2018].

Pixel accuracy, which simply computes the ratio of the number of correctly classified pixels over the number of pixels, is fairly intuitive, as it simply measures the percentage of well classified data. However, it is usually not suited to properly evaluate semantic segmentation performances, especially if the segmentation classes are unbalanced. Indeed, if we consider a dataset composed of 99% of background and 1% of targeted objects (e.g. cars), then a dumb classifier always predicting "background" has an accuracy of 99%.

Hence, like many semantic segmentation works, the metric we mostly use to

evaluate the performances in our works is the IoU, also known as Jaccard Index. It is defined as the size of the intersection divided by the size of the union of two sets. Given two sets A and B, it formally writes : $\text{IoU}(A, B) = \frac{|A \cap B|}{|A \cup B|}$. That ratio can be reformulated as the number of true positives (intersection) over the sum of true positives, false negatives, and false positives (union). It is computed individually for each segmentation class and is then usually averaged over the classes.

To investigate the refinement capabilities of our methodologies, we also look at the IoU gain with respect to the number of interactions and at the processing time.

II.C.2 . Datasets

The experiments we elaborated during this thesis have been mainly performed on five different datasets. While they all concern semantic segmentation of aerial images taken from airplanes or drones, they each have their own specificity and challenges.

— The **ISPRS Potsdam** [Rottensteiner et al., 2012] dataset is composed of 6 classes (*impervious surface, buildings, low vegetation, tree, car* and *clutter*). The class *car* is sub-represented compared to the other classes. This dataset covers around 3 km^2 with a spatial resolution of 0.05m. The size of each image is $6\,000 \times 6\,000$ pixels. The dataset is composed of 38 images.

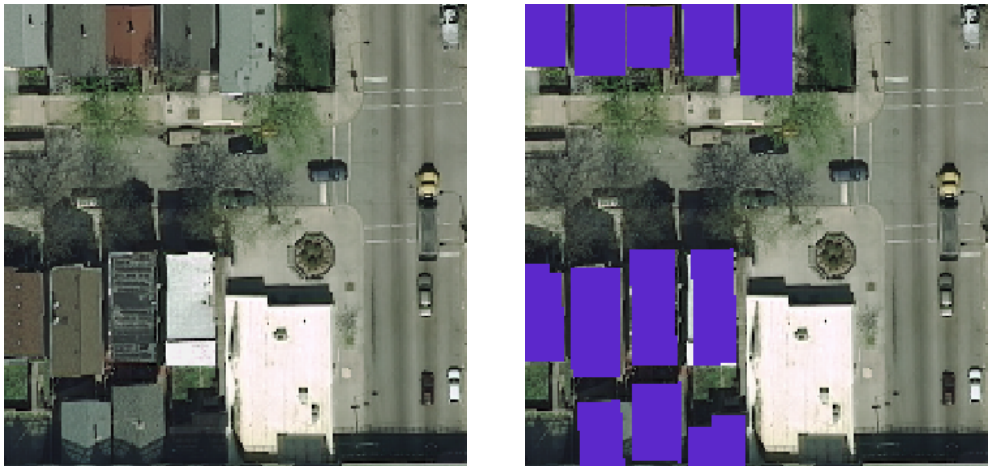
— The **ISPRS Vaihingen** dataset is composed of the same classes. This dataset is composed of 33 tiles of various sizes with 0.09m resolution. It is not RGB but IRRG.

— The **SemCity Toulouse** [Roscher et al., 2020] dataset is the smallest considered one with also the coarsest resolution. Its training set is composed of 4 3504×3452 images at 0.5m resolution and 8 segmentation classes : impervious surface, building, pervious surface, high vegetation, car, water, sport venues & void. This dataset is panchromatic and contains 8 spectral channels.

— The **INRIA Aerial Image Labelling** [Maggiori et al., 2017] dataset is composed of two classes (*buildings* and *not buildings*) and covers more than 800 km^2 with a 0.3m resolution. Its training set is composed of 180 $5\,000 \times 5\,000$ images and contains 5 different cities : Austin, Tyrol, Chicago, Kitsap and Vienna. Its labeling was automatic and is based on land register. It is thus not always signal compliant, as we can observe on Figure 2.11.

— The **Aerial Imagery for Roof Segmentation (AIRS)** [Chen et al., 2019] dataset is composed of the same two classes and covering 457 km^2 in New-Zealand at a 7.5 cm resolution.

These datasets cover many remote sensing stakes. Indeed, they address binary as well as multi-class segmentation, contain resolutions ranging from 5cm to 50cm, and are of various size. Moreover, they are not all RGB and some of them have very varied distributions. As pointed by [Schmitt et al., 2021], an equivalent to ImageNet does not exist yet in remote sensing, mainly due the multitude of sensors and task-specific models. However, thanks to their complementarity, an approach



(a) Image (b) Ground-truth

Figure 2.11 – Example of a missing building in INRIA ground-truth : the large white building is not in the ground-truth database.

validated on all the aforementioned datasets is likely to be robust to many use cases using drone and airborne data. Figure 2.12 exhibits samples from these datasets to showcase their different challenges. First, we can find in the AIRS dataset both urban areas with quite large buildings and rural areas with sparse and much smaller houses. Second, we see two peculiar situations from the Potsdam dataset. In the first line, a parking lot should be classified as "building" but the presence of cars disturb the neural network which has not seen the semantic "cars surrounded by building" during training. In the second line, a large vegetalized roof should also be classified as "building" but is mainly classified as "vegetation", which makes some sense. Then, in the INRIA dataset, the data distribution varies greatly between the different cities (e.g. Chicago and Tyrol). The first line also shows the limits of the automatically collected ground-truth : some buildings are not present in the ground-truth while an empty field is categorized as "building". Finally, we can notice that the segmentation in SemCity Toulouse is not trivial due to the large number of target classes and to the small training set. In all these examples, while the segmentations may seem accurate at first glance, an accurate inspection show that there is always room for improvement, either overall or in the details. Moreover, the errors are different depending on the images, which shows that it is difficult to put an a priori on the errors of a neural network. Therefore, these examples show the limits of current deep neural networks and we will see in the following that our proposed interactive methodologies can be a way to overcome them.

Except when specified otherwise, all of our experiments use the different datasets in a similar fashion. We split the initial training sets into a training and a validation sets with a 80%-20% ratio and use the validation sets to evaluate our refinement methods.

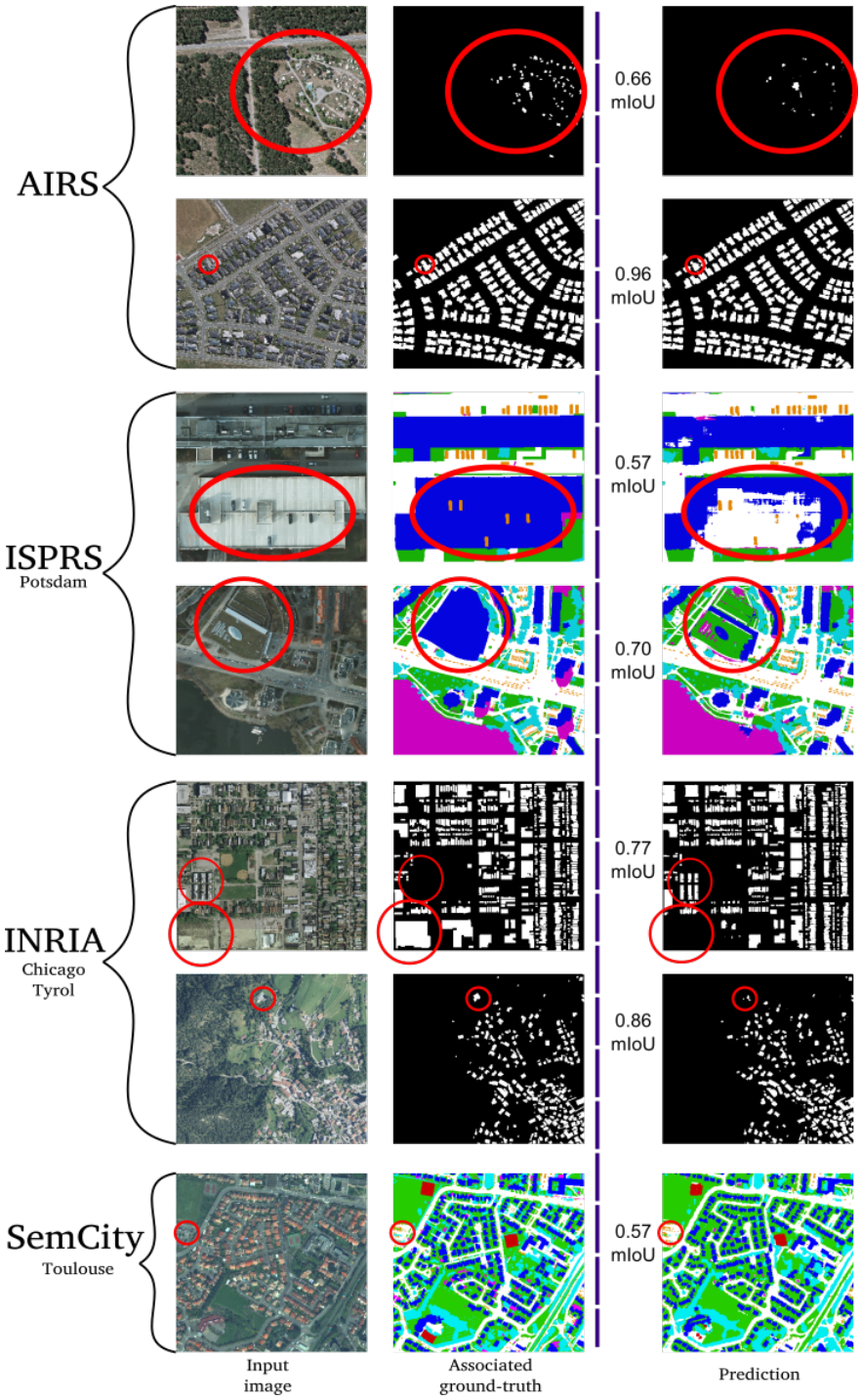


Figure 2.12 – Result samples of a U-Net on standard datasets. The red circles represent peculiar wrong predictions, with some explained in the text. Best viewed in color.

Chapter III - Fast interactive learning

Contents

III.A . Motivation & contribution

With abundant and well-annotated training data, deep neural networks can learn to perform a specific task very well. Hence, when trained for semantic segmentation, they are able to produce accurate segmentation maps and lack only a few percents of precision to reach perfect scores on public benchmarks. However, these few percents and slight mistakes can visually make a big difference and therefore not be tolerable in practice.

In order to concretely motivate our approach, we consider two of the aforementioned aerial image datasets. On the INRIA Aerial Image Labelling Dataset [Maggiori et al., 2017], the current best networks reach an IoU around 0.8 and a pixel accuracy around 97% on the test set. On the ISPRS Potsdam multi-class segmentation dataset [Rottensteiner et al., 2012], the state-of-the-art approaches almost reach a pixel accuracy of 92% on the test set. While these performances are incredibly high, there still remains some misclassified areas, as previously shown on

Figure 2.12, that would potentially be unacceptable for an end-user. Besides, these optimal results are obtained using top notch neural networks which have required many specific refinements [Yue et al., 2019]. An off-the-shelf neural network and training strategy still yield good results but, as the baselines coming along the datasets show, a drop of performance between 5 and 10% can be expected, which corresponds to perceptibly wrong segmentation maps.

We also consider a practical application for which current approaches still yield imperfect results. Drones are increasingly used to monitor different environments like crop fields, railroads or quarries. In this context, semantic segmentation can be extremely useful for different tasks such as defect detection, volume computation or crop monitoring. However, due to the complexity and the high variety of the acquisitions often obtained in changing conditions, results of a given model are usually not as good as on public datasets while a high precision is necessary for these tasks. Therefore, the operators often have to manually refine the segmentation maps to meet customer requirements, which is a slow process.

To address these issues, we propose to adopt an incremental and interactive semantic segmentation approach, as sketched in Figure 3.1. Starting from a segmentation map initially proposed by a neural network, the user indicates the mistakes to the algorithm, which then uses these annotations to correct the errors. Indeed, a human in the loop can easily spot the misclassified areas them thanks to a more complex yet intuitive analysis. The difficulty then is to reach optimal classification while keeping the whole process swift and engaging enough. This is why we propose an approach without any retraining component, using the annotations to modify only the inputs of the neural network.

Most of the works presented in this chapter have been published in **DISIR : Deep Image Segmentation with Interactive Refinements** Lenczner G., Le Saux B., Luminari N., Chan-Hon-Tong A. & Le Besnerais G., ISPRS Annals 2020 and the associated code is available on this GitHub repository : https://github.com/delair-ai/DISIR.

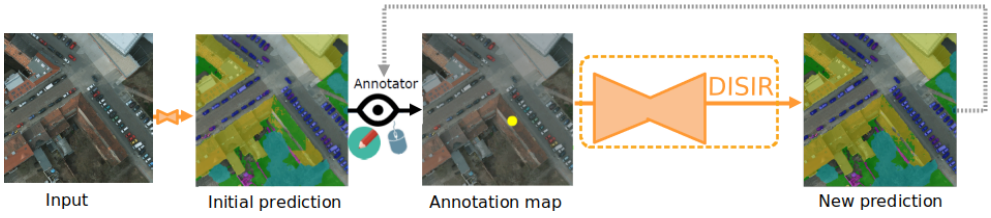


Figure 3.1 – Visual representation of DISIR. The framework starts with an initial prediction that the annotator can annotate with new points to fix errors. Best viewed in color.

III.B . DISIR : Deep Image Segmentation with Interactive Refi-

nements

We now describe in details the proposed approach for interactive N-class segmentation of aerial images. In particular, our goal is to train a neural network combining two criteria :

1. **Semantic segmentation :** The neural network is able to provide an initial accurate segmentation map of the scene without any additional help.
2. **Interactive refinement :** The neural network can also use annotations provided by an operator to efficiently fix its mistakes and quickly enhance its initial prediction.

To achieve this, we propose a neural network that keeps its original structure but takes as input a concatenation of the classic inputs (e.g. RGB) and of the annotations (N channels, one per class). These annotations are clicked points. When there are no annotations, the annotation channels are initially filled with zeros. Only the inputs of the network are modified and not its weights, which makes the swiftness of the approach.

We first define our training strategy and then present our study on the annotations themselves.

III.B.1 . Training strategy

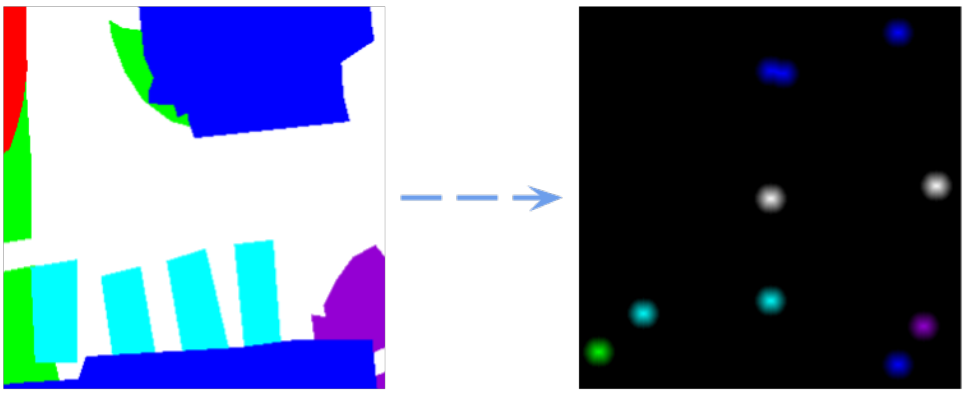


Figure 3.2 – Annotation sampling during training from a ground truth map. In practice, these annotations are encoded in the N annotation channels

During training, the neural network needs to learn how to use the clicked points as guidance to enhance its initial predictions. With this aim, ground-truth maps are the core of our training strategy. On the one hand, they are classically used to compute and back-propagate the loss. On the other hand, as illustrated on Figure 3.2, they are also *randomly sparsified to sample annotations*. In other words, only a few pixels from the ground-truth are kept to be used as annotations. According to their class, these annotations are encoded in the N annotation channels

given as input to the algorithm. To train under various annotation layouts, the number of sampled annotations is random in each training example. Image-only inputs are also sampled to train segmentation in a standard way and ensure that the network proposes accurate initial segmentation maps.

If the annotations are sampled independently of their class, the following problem may occur. During the evaluation phase, annotations on sub-represented classes can be ignored by the network because it has barely seen any annotation points of these classes during training. Therefore, it has not learned how to use them to enhance its predictions. To overcome this issue, we use a *frequency balancing* strategy to sample the annotations based on the classes distributions. It allows the network to equally see annotations from each class during training and, therefore, to be efficiently guided once the training is done.

III.B.2 . Annotation representation

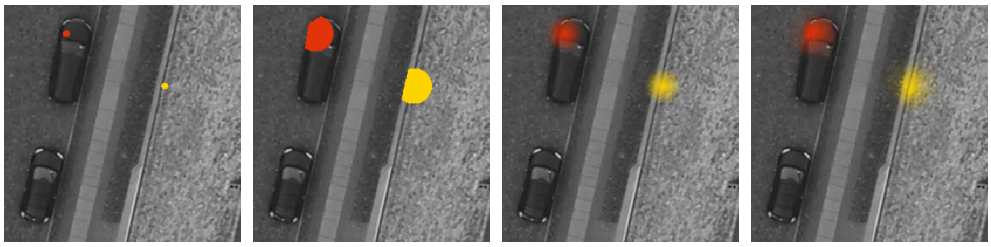
We investigate two aspects of the annotation representation : how to *position* clicks in order to sample the most useful information, and how to *encode* clicks to get the best benefit.

III.B.2.a Click positioning.

Fixing a wrong segmentation implies to provide the system with additional information about the right division. New samples provided by clicks may represent either the inside of an instance or its border.

The first case seems to be the most intuitive. Clicked pixels are inside instances and the annotation points represent the class associated to these instances. Contrary to [Xu et al., 2016], we do not sample them at a minimal distance from the boundaries since we assume that an annotator might click near an edge to fine-tune the prediction. For the second case where the annotations represent the borders of the instances, the channel associated to a click corresponds to a class randomly chosen among the ones adjacent to the clicked border.

Aiming to ease the burden of the end users, we also explore softer constraints on the annotations. Indeed, instead of using N annotation channels, we summarize them into a single annotation channel. For the border strategy, this single channel only indicates the presence of a border. For the inside point strategy, it only indicates where the network has initially made a mistake. To implement this latter strategy, we have to slightly modify the training process. The network performs a first inference to create a segmentation map used to find mislabeled regions. Annotations are then sampled in these areas and a second inference is performed. Only this second inference is used to back-propagate the gradients. However, as shown in Section III.D.3.a, none of these simplified annotations seems promising to efficiently guide the segmentation task.



(a) Car (red) and building (yellow) annotations

(b) Ground-truth connectivity (C-GT) encoding

(c) Distance-transform (DT) encoding

(d) Guided filter (GF) encoding

Figure 3.3 – Different annotations encodings depending on context uses. Best viewed in color.

We investigate here the annotations encoding to analyze its influence on DISIR.

There are many possibilities to encode the annotations in their dedicated channels and they all provide different spatial information. The size of the encoding is the most obvious issue : if it is too small, it might not provide enough information to efficiently fix the initial segmentation but a coarser encoding might provide erroneous information. A popular context-free trade-off used in most interactive segmentation works [Xu et al., 2016, Liew et al., 2017] is to encode the annotations with Euclidean distance transforms to dilute spatial information. However, due to its context independence, this encoding might be sub-optimal. Ideally, the perfect encoding would be the original ground-truth map but it is obviously impossible to get. Based on this insight, we study here how to best approximate this ground-truth given the available data : the input image, the annotations and the trained neural network. We define two possible context use besides the no-context one :

— Using the input image
— Using the initial prediction

We therefore propose to study five ways of encoding annotations :

— A first encoding baseline consisting of small binary disks (bin.).
— A second baseline consisting of a distance transform (DT) applied on larger disks.
— To use the input image, we rely on guided filtering [He et al., 2010] (GF) in order to preserve the edges in the encoding.
— To use the initial prediction, we encode the annotations using their connected pixels in the prediction map (C-PM).
— To estimate the superior boundary theoretically reachable with an encoding from the ground-truth, we also encode the annotations using their connected pixels in the ground-truth map (C-GT).

These different encoding methods are represented in Figure 3.3.

III.C . Evaluation process

We now briefly summarize the context of our methodology. A user needs to quickly and accurately semantically segment Earth observation images. This user has also access to another annotated database which, depending on the use-case, may or may not belong to the same domain as the targeted images. For the sake of simplicity, the annotated label space must be the same as the targeted one. We propose to first train a neural network on the annotated database. Then, the user can use this neural network to make predictions on the target images. If the segmentation result is not accurate enough for the user's requirements, they can interact with the network to refine its predictions. These user interactions come in the form of clicked points on the mislabeled areas and represent their corresponding labels, as chosen by the user.

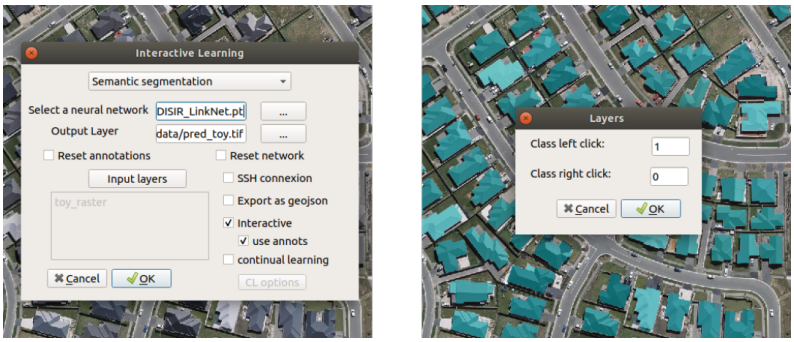


Figure 3.4 – Minimalist interface of our QGIS plugin for semantic segmentation and interactive refinements

The evaluation of this kind of process is not standard since usual semantic segmentation methods are fully automatic. Hence, we have developed a QGIS * plugin available with the code † to allow potential users to experience the proposed framework. In this plugin, user interaction is handled by the QGIS interface (presented on Figure 3.4) while the heavy computations, e.g. the semantic segmentation, are performed in a separate server that can be local or remote. Once the server is launched, the data transfer is transparent to the user. However, this is not suited to conduct a large scale evaluation as it would require many users to get statistically relevant results. Hence, we have simulated the user behavior to automatically generate interactions. In this case, the annotations are sampled in the mistake areas using a comparison between the ground-truth map and the prediction map. This automatic evaluation thus necessarily requires a partial access to the ground-truth maps, even though **they are only used to sample annotations**.

Moreover, we have compared two click simulation strategies, both using a comparison between the prediction and the ground-truth :

1. A click is automatically sampled in one of the biggest mislabeled areas. Some randomness is added in the choice of the area and in the localization

∗. http ://qgis.osgeo.org
†. https ://github.com/delair-ai/DISIR

of the click inside the area to better simulate a human behavior. This process is class-independent.

2. The process is similar to 1. but the generated click has the supplementary constraint to be on a pixel belonging to a specified class. This allows to also correct pixels belonging to sub-represented classes. This process is class-dependent.

As we will see in Section III.D.2, the class-dependent process is better than the first one to evaluate the influence of the clicks on sub-represented classes but has two drawbacks : overall smaller erroneous areas are corrected which leads to smaller corrections and there is less room for randomness if the chosen class to annotate is predetermined.

In the rest of this thesis, we refer both to the synthetic operator and to the potential human user as *the agent*.

III.D . Experiments

In this section, we aim to show that our method works and study the influence of the different parameters. Furthermore, we conduct three experiments to better apprehend the possibilities and the limits of our approach :

— We first compare different backbone architectures to evaluate if it has a significant impact on the performances. More importantly, since these different architectures produce different initial segmentation maps, this comparison also allows us to study if the initial quality of the segmentation maps influences the benefits brought by the annotations.

— The second one is motivated by the fact that it often happens in practice to only have access to a very limited amount of annotated data. Similar to [Castillo-Navarro et al., 2019] where the authors study the influence of the training set size on the network performance, we study the influence of this parameter on the neural network refinement abilities. To this end, we have trained the networks on subsets of the initial training sets.

— Finally, since remote sensing datasets can have varied spatial resolutions, we evaluate the influence of this factor on our approach by progressively degrading it.

This section is thus organized as follows. We first present our experimental setup in III.D.1. Second, we show that our method works and how to best evaluate it in III.D.2. Then, in III.D.3, we analyze with automatic evaluation the influence of the following parameters : the annotation strategy, the annotations encoding, the network backbone, the volume of training data and the pixel resolution. Finally, we draw conclusions from the evaluations with human operator in III.D.4.

III.D.1 . Experimental set-up & hyper parameters

We experiment on the INRIA Aerial Image Labelling dataset and the ISPRS Potsdam dataset.The initial training sets are divided into a smaller training set and a validation set with a ratio 80%-20%. The ground-truth availability of the validation sets allows to synthesize the annotations required to automatically evaluate the framework. The images are tiled into patches of size 512×512 with an overlap of size 128 to be processed.

Except in the backbone comparison, we use a LinkNet [Chaurasia and Culurciello, 2017] architecture. The networks are trained using stochastic gradient descent (SGD) and cross-entropy loss for 50 epochs with a batch of size 8, seeing during each epoch $10\,000$ samples randomly chosen and cropped (size 512×512). The initial learning rate is fixed at 0.05 and is divided by 10 after 15, 30 and 45 epochs. Only basic data augmentation is performed : horizontal and vertical flips. The implementation is done using Pytorch.

Except when specified otherwise, the annotations are encoded into the neural network channels inputs using distance transform.

III.D.2 . Approach assessment

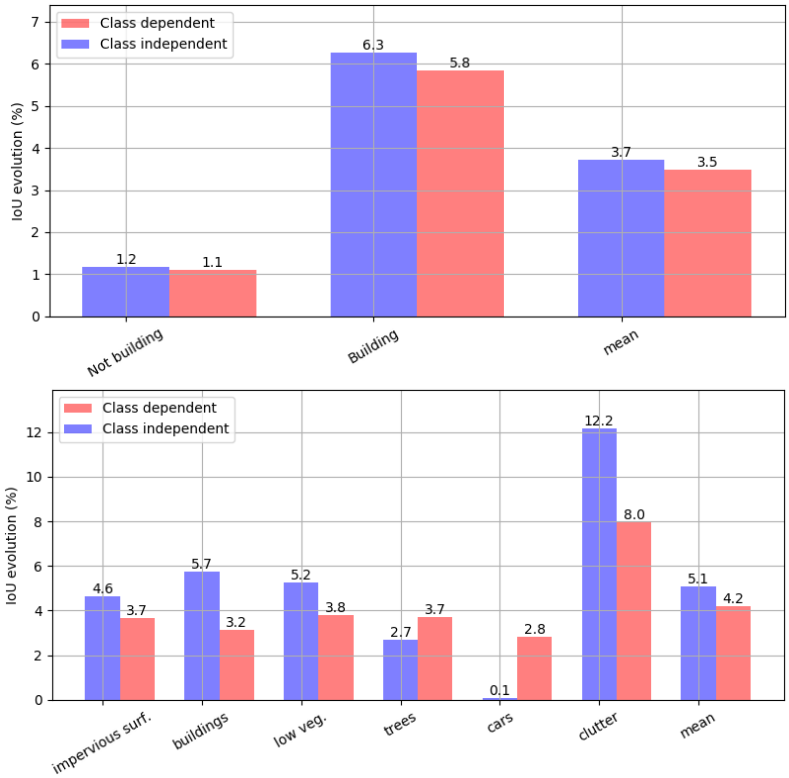


Figure 3.5 – Comparison of the two different automatic evaluation processes with 120 annotations on the INRIA (top) and Potsdam (bottom) datasets.

We first assess the proposed method with the two proposed automatic evaluation strategies. Let us recall that, for the first strategy, the clicks are sampled iteratively

in the biggest erroneous areas independently of the class while they are sampled equally in each class in the second one.

As we can see on Figure 3.5, the class independent evaluation strategy allows to reach a higher overall IoU but to the detriment of the sub-represented *car* class in the Potsdam dataset. This is due to the fact that the biggest erroneously predicted areas inherently belong to larger instances than cars such as buildings. Therefore, even though the overall metric gain is not as good as with the class-independent evaluation process, we choose the class-dependent one for our further evaluations on Potsdam. However, since the INRIA dataset does not contain a low-represented class, we choose the class-independent evaluation process to evaluate our experiments on this dataset.

Whatever the evaluation strategy, the results displayed in Figure 3.5 validate the efficiency of our approach on both datasets. Indeed, all segmentation performances are improved for all classes : on average, the mean IoU is increased by 3.7% on the INRIA Building dataset, and by 4.2% on the muti-class ISPRS Potsdam dataset for 120 clicked annotations. Besides, as we can see on Table 3.1, each click allows to correct around $5\,000$ pixels in average.

We have shown that both evaluation methods (class-independent and class-dependent) yield the same trends. In the following, we use class-dependent evaluation only for imbalanced datasets such as ISPRS Potsdam.

Dataset	Corrected pixels
INRIA	3143
Potsdam	7219

Table 3.1 – Average corrected pixels per click

III.D.3 . Influence of the different parameters

III.D.3.a Influence of the annotation strategy

We now compare the different annotation strategies : inside clicks or border clicks, and single or multiple channels. We also compare our two encoding baselines that are binary encoding or distance transform.

As we can see on Figure 3.6, the distance transform slightly increases the benefits of the annotations compared to the binary encoding. While [Benenson et al., 2019] conclude that the binary encoding leads to better performances, our opposite conclusion might be inherent to the large size and scale of aerial images which dilute the annotations localized over very small areas.

Regarding the localization of the annotations, both the contours and the inside points are efficiently used by the network to enhance its predictions but it is still noticeably better with the inside points. We can also notice that the last 20 added points considerably boost the performances of the inside point strategies for the Potsdam dataset : this is due to the fact that these points belong to the class

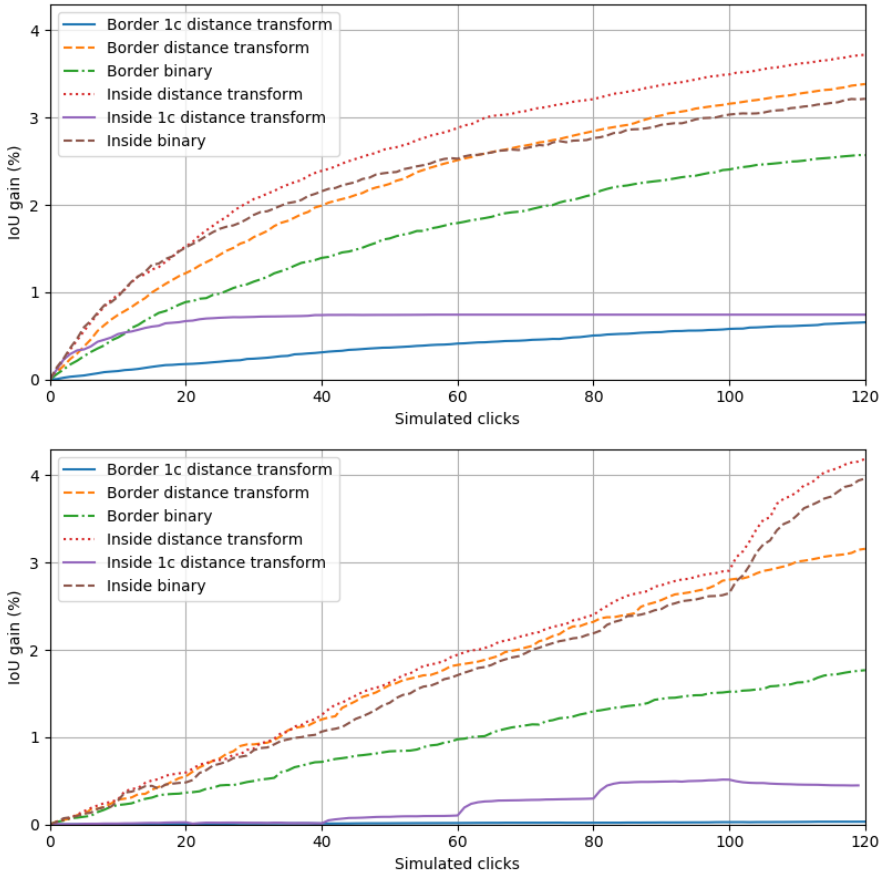


Figure 3.6 – Comparison of the different annotation strategies on the INRIA (top) and Potsdam (bottom) datasets. *1c* means single annotation channel.

clutter, an under-represented class. Therefore, they have the strongest impact in term of IoU, even though this would not be the most important class to correct for a human user.

Finally, the two degraded strategies which rely on single annotation channel bring little or no improvement even though it is slightly better for the INRIA dataset since it contains only two classes.

Therefore, considering these results, we only use the inside annotation strategy with multiple annotation channels in the remainder of this thesis.

III.D.3.b Influence of the annotations encoding

We then delve into the details of the annotations encoding. As shown in Table 3.2, the different encoding strategies seem to provide similar information to the network as the gains are in the same order of magnitude. Indeed, they all increase the IoU of around 6% for 120 annotations on the ISPRS images, even though the binary encoding is slightly lower and confirms the usefulness of Distance

	Bin.	*DT*	*C-PM*	*C-GM* (sup)	*GF*
Initial	70.7	70.7	70.7	70.7	70.8
After	76.4	76.6	76.5	**76.7**	76.7
Gain	5.7	5.9	5.8	**6**	5.9

Table 3.2 – IoU on ISPRS after 120 annotations with DISIR depending on the encoding.

Transform (DT) encoding. The Guided Filter (GF) encoding obtains the same gain as the DT one, the Connected in Prediction Map (C-PM) encoding is lower of 0.1% and even the golden standard (i.e. the Connected in Ground-truth Map (C-GM) encoding) is only better of 0.1%.

These insignificant differences show that the network does not need any contextual guidance to learn nearly optimal information from the annotations using a simple and intuitive encoding such as distance transform.

The slight superiority of distance transform over binary annotations make us favor this former encoding in the remainder of the thesis.

III.D.3.c Influence of the network backbone

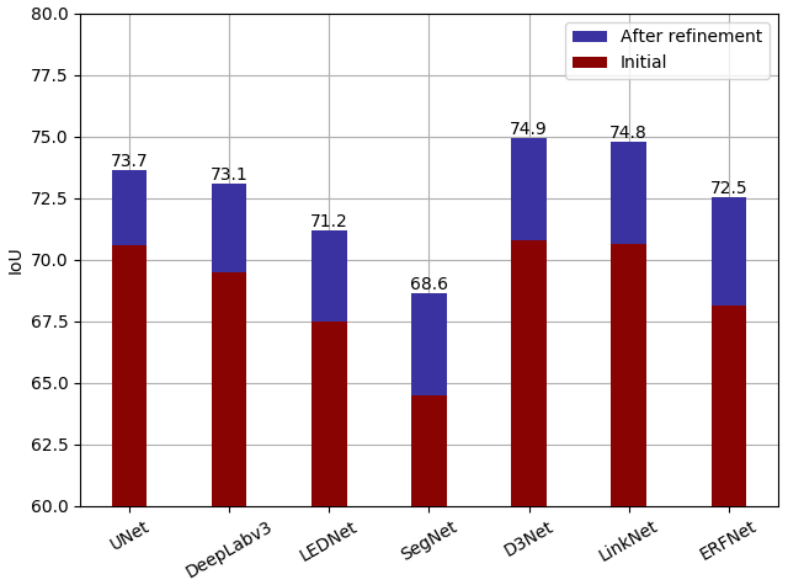


Figure 3.7 – Study of the impact of the architecture choice on the Potsdam validation set sorted per IoU gain.

We compare LinkNet to SegNet [Badrinarayanan et al., 2017], UNet [Ronneberger et al., 2015] and DeepLabv3 [Chen et al., 2017] which are standard segmentation networks

with increasing complexity and also to the following lighter architectures : LED-Net [Wang et al., 2019a], ERFNet [Romera et al., 2017] and D3Net [Carvalho et al., 2018]. Figure 3.7 shows the results obtained with the different architectures under the same training and evaluating conditions. As expected, the gains are in the same order of magnitude. Indeed, the initial IoU mean is 68.8% with a standard deviation of 2.13 while the IoU gain mean is 3.9% with a standard deviation of 0.4. Figure 3.7 also shows that the accuracy gain of the interactive correction seems to be uncorrelated to the accuracy of the initial segmentation map. For instance, the worse initial architecture here – SegNet – is the average one in regards to the IoU gain.

We have shown that this approach is effectively agnostic to the network architecture. Even though we use LinkNet for most of our experiments, any convolutional neural network could work with DISIR.

III.D.3.d Influence of the volume of training data

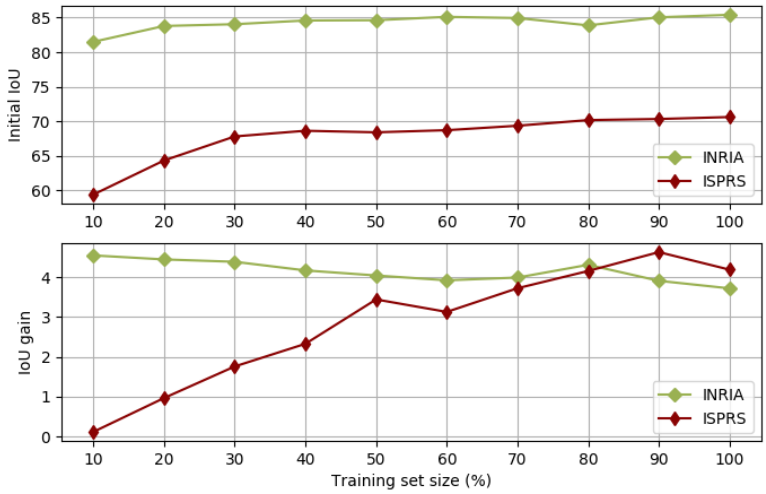


Figure 3.8 – Influence of the training set size on the initial IoU (top) and on the IoU gain (bottom)

Figure 3.8 shows the influence of the training set size on our approach. The different behavior on the two datasets can be explained by their initial size difference.

On the INRIA dataset, since the initial training size is high, even 10% of the training data seems to be enough to provide a network with a decent initial accuracy and a good ability to use the annotations. Besides, if more data implies a better initial accuracy, it does not improve the performances of the interactive correction. This shows that the network has not learned to make a better use of the annotations with supplementary data. The plateau reached by the algorithms on this dataset is possibly due to the ground-truth noise inherent to the INRIA land register-based labeling.

On the Potsdam dataset, even though the initial training size is lower than in the INRIA dataset, the network is still initially quite accurate with little training data. Indeed, according to the results of [Castillo-Navarro et al., 2019], since there are pictures from only one city, few training images are enough to learn the general semantic of the dataset even if the full training set provides better performances. However, the accuracy gain is really low with little training data. For example, the IoU gain is less than 1% with 20% of the initial volume of data while it is slightly over 4% with the full training set. We believe that this lack of performance in low-data regime is due to over-fitting. Indeed, since there are only a few training images in this scenario, there are also less possible annotations and they might not fully reflect the reality of the test set. Besides, if the network over-fits on these few images, it might also consider the annotations as unnecessary for the segmentation.

Therefore, as shown by the study on the Potsdam dataset, a certain amount of data seems necessary to optimally use the annotations. However, as shown by the study on the INRIA dataset, the network ability to use the annotations reaches a plateau once there is enough available training data.

Unfortunately, this outcome is verified on the SemCity Toulouse dataset where we fail to improve the segmentation results using DISIR, even considering only the binary case of building segmentation. Since the entire annotated dataset is composed of four 3504×3452 images, we have approximately the same amount of annotated pixels if we consider three out these four images for the training set than with a single image from the ISPRS Potsdam dataset ($3.6e^{10}$ pixels).

III.D.3.e Influence of the pixel resolution

In addition to being the smallest, the SemCity Toulouse dataset is also the one with the lowest pixel resolution (50cm/pixel) of those we work with. To ensure that this factor is not another bottleneck for DISIR, we degrade the resolution on AIRS from 7.5cm/pixel (original resolution) to 50cm/pixel (SemCity Toulouse resolution). We do not perform this experiment on ISPRS Potsdam because we would again deal with a lack of training data when reducing the resolution to 50cm. As we can observe on Figure 3.9, although the initial performances vary, partly due to the stochastic nature of the experiments, DISIR consistently improves segmentation maps by 3%-4% IoU with 10 annotations regardless of pixel resolution.

This shows that pixel resolution does not heavily influence DISIR, at least between 5cm/pixel and 50cm/pixel. Hence, the lack of training data is clearly the bottleneck on SemCity Toulouse and is a main limitation of DISIR.

III.D.4 . Analysis with human operator

In this experiment, the images from the Potsdam validation set have been manually refined by a human annotator. If the number of clicks exceeds 120, we threshold it at 120 in order to make a fair comparison with the automatic process.

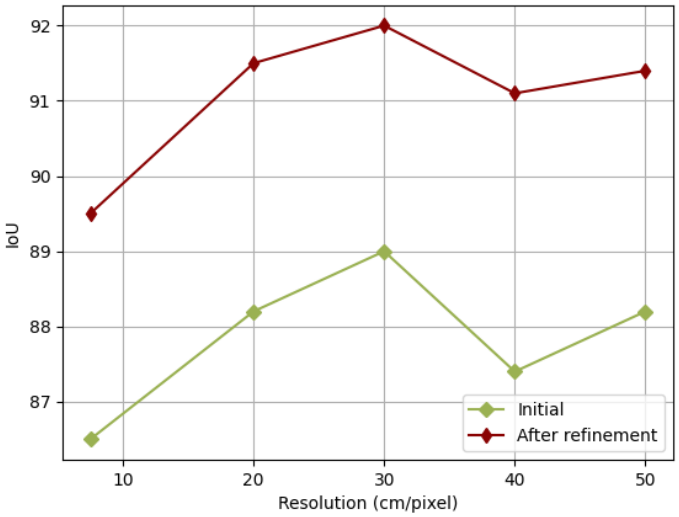
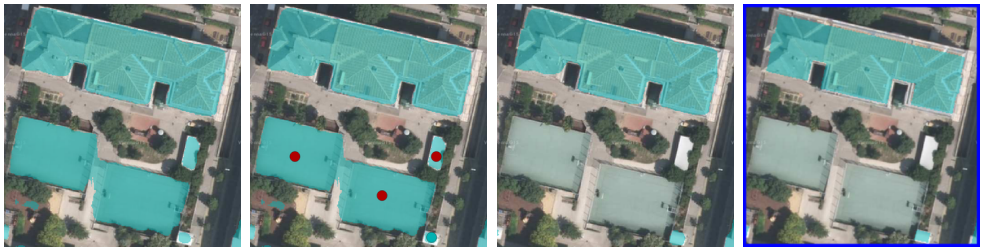


Figure 3.9 – Influence of the pixel resolution on DISIR on the AIRS dataset.

III.D.4.a Local insights.

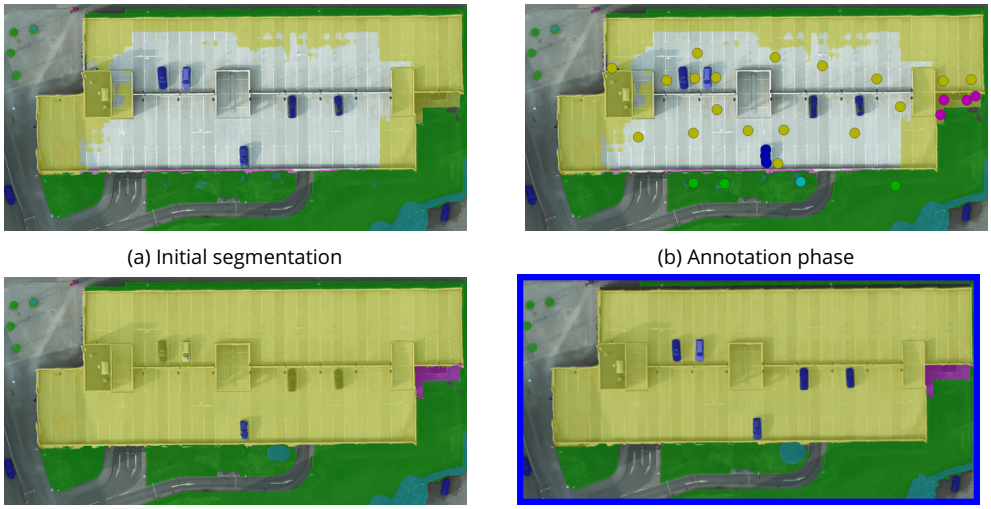
On the one hand, as shown on Figure 3.10, the refinements can be very intuitive and effective on areas semantically similar to the ones seen during training. On the other hand, if the semantic is new compared to what is in the training set, the neural networks have trouble to use the annotations efficiently.



(a) Initial segmentation (b) Annotation phase (c) Refined segmentation (d) Ground-truth

Figure 3.10 – Annotations lead to an easy false positive buildings removal on the segmentation map

For example, in the Potsdam dataset, there is only one outside car park considered as building which means only one place with the semantic "*car*" surrounded by "*building*" in the dataset. We kept the associated image in the validation set to study the impact of the annotations in this scenario. Figure 3.11 shows the outcome of our approach on this car park. Since it also looks like a road, it is initially difficult for the network to segment it correctly. Nonetheless, it succeeds



(a) Initial segmentation (b) Annotation phase

(c) Refined segmentation (d) Ground-truth

Figure 3.11 – Difficult segmentation of an outside parking since the network has not learned the semantic *cars on building*. Only the car at the bottom is annotated and thus recognized.

to recognize the cars parked there. Then, with building annotations, the network successfully recognizes a building. However, it also considers that the vehicles parked there are now part of the building since it has never seen the class "*car*" surrounded by the class "*building*" during training. As we can see on Figure 3.11, with supplementary *car* annotations, the network can still recognize the correct semantic of the scene. However, the process in this case is not very smooth and intuitive since the cars which were primarily well recognized need to be annotated nonetheless. This example shows that our framework does not perform optimally when it is faced to areas with a different semantic compared to the ones present in the training set.

Finally, Figure 3.11 also shows the locality of DISIR. Indeed, even though the network is able to perform local corrections with the annotations, it is not able to propagate the information of these annotations on a large scale in the image. Indeed, it is limited both by the receptive field of the neural network and by the size of the processed patches.

III.D.4.b General insights.

Regarding the click distribution, as shown in Figure 3.12, a human operator tends to focus clicking on specific areas while the automatic evaluation rather spreads the annotations all across the image. However, as shown in Figure 3.13, these grouped clicks seem to efficiently increase the metric. Indeed, with the manual evaluation, 4 classes out of 6 are more improved and the mean IoU gain is overall better. This shows the efficiency of our approach with a real user in the loop.

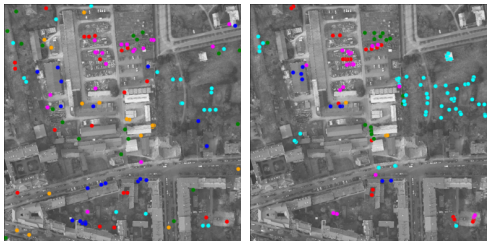


Figure 3.12 – Click distribution on an image from Potsdam in automatic (left) and manual (right) evaluations. The colors represent the different classes.

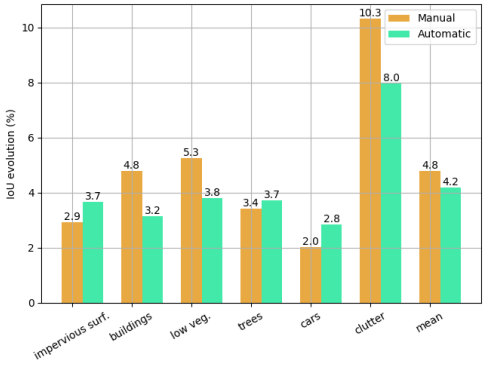


Figure 3.13 – Comparison of the IoU evolution between an automatic (green) and a manual (gold) evaluation on the ISPRS dataset.

However, the simulated process still seems a good proxy to evaluate the approach. Finally, Figure 3.14 shows qualitative results before and after human interaction.

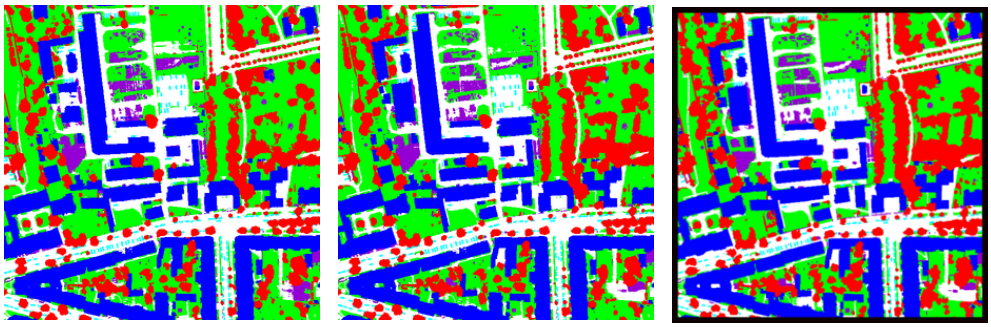


Figure 3.14 – Full predictions and their accuracy before/after human interaction (HI) on an image from Potsdam

III.E . Conclusion

We have proposed in this chapter an interactive multi-class segmentation framework for aerial images. Starting from a neural network designed for semantic segmentation purpose, it consists in training this network to exploit user annotation inputs. At testing time, user annotations are input in the neural network without changing the parameters of the model, hence the interactive semantic segmentation process is swift and efficient. Through experiments on public aerial datasets, we have shown that interactive refinement is efficient for all classes for datasets ranging from two to six segmentation classes. It improves classification results by 4% on average for 120 clicks and, mainly, produces segmentation maps which are visually

more rewarding. We have shown that our interactive process is efficient whatever the network backbone is. We have also investigated different representations of the annotations and have concluded that clicks positioned inside instances and encoded using distance transform carry the most meaningful information. We have seen that the encoding strategy does not greatly modify the results. Indeed, the network is able to apprehend the context of the annotation from the annotation and the image alone, without additional filters on the annotation encoding. However, while the spatial resolution has no impact on the efficiency of the proposed method, a lack of training data prevents the neural networks from learning to be guided by annotations. This is particularly a problem for the SemCity Toulouse dataset which contains relatively little annotated data.

The other identified main limitation of a network trained with DISIR is the spatial propagation of the information provided by the annotations, as it is not able to propagate this information far from the corresponding annotation. We address this phenomenon in the following Chapter.

Chapter IV - Interactive learning at scale

Contents

IV.A . Transformers for a better propagation of the annotations

IV.A.1 . Motivation

With DISIR, the corrections are spatially limited around the annotations. Since remote sensing images are too big to be processed entirely by deep neural networks and thus need to be tiled to be processed by patches, this spatial limitation stems from two distinct problems. On the one hand, the limited receptive field of convolutional neural networks constrains the propagation of information within the processed patch. On the other hand, the size of the processed patch constrains the propagation within the whole image. We call these problems respectively *intra-patch propagation* and *inter-patch propagation*.

The latter issue can be addressed either by processing bigger patches or by a retraining step. We will consider a retraining strategy in Section IV.B but this necessarily lengthens the processing time and bring potential instability due to the retraining. Processing bigger patches then appears as an ideal solution. However, besides the computational challenges, this solution obviously aggravates the intra-patch propagation issue. To deal with problem, we decided to look into solutions to

enlarge the neural networks receptive fields and we notably focused on Transformer architectures and the self-attention mechanism.

IV.A.2 . Self-attention mechanism

Coming from Natural Language Processing (NLP), self-attention modules [Vaswani et al., 2017] compute the representation of each position by a weighted sum of the features at all positions. This allows to capture long-range relations, at the penalty of the computation cost scaling quadratically with the number of pixels. These modules take usually as input an embedding of the raw input (e.g. the output of another self-attention layer or from an initial embedding layer). Each input (e.g. a word in NLP or a group of pixels in computer vision) is referred to as a *token*.

To get into details, a self-attention module is composed of three learnable vectors called *query* q, *key* k and *value* v. For a token x_i, an attention weight s is computed at each position between this token query and the other keys : $\forall j, s_{i,j} = q_i \cdot k_j$. These attention weights are secondly divided by the square root of the dimension of the key vectors, $\sqrt{d_k}$, to stabilize the gradients : $s_i = \frac{s_i}{\sqrt{d_k}}$. They are then softmaxed and multiplied by the other token values : $s_{i,j} = \text{softmax}(s_i)_j \cdot v_j$. This keeps the values of the token of interest, and drowns-out irrelevant ones. Finally, the weighted value vectors are summed to produce the output of the self-attention layer.

In practice, this calculation is done in matrix form for performance reasons. Hence, given Q, K and V respectively the query, key and value matrices, we can formally write :

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_K}}\right)V$$

Moreover, the token positions (e.g. this group of pixel is at the top left of the image) are encoded in the initial embedding and each self-attention module is actually composed of multiple self-attention head (i.e. key, query and value matrices) to enable more diversified latent representations.

IV.A.3 . Transformer architectures : related work

Transformers mainly rely on self-attention blocks combined with MLPs and normalization layers. After their resounding success in NLP [Vaswani et al., 2017, Devlin et al., 2019], these architectures have recently started to be applied to computer vision by considering blocks of pixels as visual words. The ground-breaking Vision Transformers (ViT) [Dosovitskiy et al., 2021] first addressed image classification with a pure Transformer architecture, reaching good results under the condition of large pre-training. SETR [Zheng et al., 2021] extends it to semantic segmentation with simple convolutional decoders. SegFormer [Xie et al., 2021] improves this baseline with a multi-scale pure Transformer architecture. However, despite their proven feature extraction abilities, these architectures are difficult to train due to the

$O(N^2)$ computation cost of the self-attention layers and the lack of prior inherent to convolutional architectures. Hence, some works like SWIN [Liu et al., 2021] propose alternative and less expensive ways to compute self-attention. Different works also propose to combine transformer modules and convolution ones for semantic segmentation. TransUNet [Chen et al., 2021a] combines them sequentially with initial convolution blocks being followed by transformer blocks while TransFuse [Zhang et al., 2021] uses a transformer branch and a convolutional one in parallel to benefit from both representations. We mostly rely on such hybrid architectures in our works.

IV.A.4 . Methodology

To analyze the impact of the receptive field on the propagation of the annotation within a patch, we look at the extent of modified pixels using DISIR from a single clicked annotation. The goal is to verify if a transformer architecture allows to propagate further and better the information of an annotation than a convolutional architecture.

We first compare three purely convolutional networks : a light LinkNet with a ResNet34 encoder, an heavier DeepLabv3+, which relies on atrous convolutions to enlarge its receptive field, with a ResNet101 encoder and a UNet with an EfficientNet-b0 [Tan and Le, 2019] encoder (referred to as Unet-eff), an optimal architecture resulting from neural architecture search. We also study a pure transformer network, Segmenter [Strudel et al., 2021], relying on the smallest ViT encoder. We finally explore in-between architectures with TransUNet and, inspired by the ideas of TransFuse, we add transformers branches in parallel with convolutional branches in LinkNet and UNet-eff that we respectively call TransFuseLinkNet and TransFuseUnet-eff. Since the addition of these branches is computationally expensive, we could not do it for DeepLab or larger versions of EfficientNet like EfficientNet-b7.

IV.A.5 . Experiments

IV.A.5.a Propagation of a single annotation

To evaluate the impact of the transformer component, for each test dataset, we sample $10\,000$ 512×512 patches, make an initial inference, generate an user annotation inside a mistake area and make a second prediction using this annotation as guidance. Then, we measure the following mean metrics and their associated standard deviation over the $10\,000$ samples :

- The initial accuracy or IoU.
- The metric gain with the generated annotation
- The spatial impact of the annotation (i.e. how far is furthest modified pixel from the annotation and how far are in average the modified pixel from the annotation).

As we can observe on Table 4.1, Segmenter reaches the lowest initial results

	Accuracy		Corrected area	
	Initial	*Gain*	*Mean*	*Max*
LinkNet34	86.9$^{+- 0.3}$	0.82$^{+- 0.06}$	127	341
Unet-eff	87.4$^{+- 0.2}$	0.84$^{+- 0.07}$	198	476
DeepLabv3+	88.2$^{+- 0.2}$	0.73$^{+- 0.12}$	181	471
TransUNet	84.9$^{+- 0.3}$	1.27$^{+- 0.08}$	228	499
TransFuseLinkNet	85.7$^{+- 0.3}$	1.55$^{+- 0.14}$	229	497
Segmenter	65.2$^{+- 0.25}$	6.8$^{+- 0.03}$	253	507
TransFuseUnet-eff	88.7$^{+- 0.16}$	0.3$^{+- 0.03}$	182	465

Table 4.1 – Accuracy gain and spatial impact of an annotation depending on the DNN architecture on the ISPRS Potsdam dataset with 512×512 patches.

(approx. 20% less than the others) and it does not catch up using the annotation. These low results can be explained by the fact that we use the smallest proposed architecture. In any case, our difficulty in making this architecture converge made us favor experiments with in-between architectures for the other experiments.

Regarding the fully convolutional architectures, LinkNet has a lower initial accuracy than the others (approx. 1%) but similar gains (approx. 0.8%). Interestingly, the corrected pixels are sensibly closer to the annotation than with the others. Its Transformer counterpart has a lower initial accuracy (-1.2%) but the annotation allows a relatively important gain (+0.73%). Moreover, the transformer branch clearly enhances the propagation of the annotations : they almost double the range of the network.

On the other hand, DeepLab and Unet-eff have high initial accuracy and already enable a good propagation of the annotations, thanks to their large receptive field. Hence, the transformer counterpart of Unet-eff does not enable a better propagation of the annotations, even though it enhances its initial performances.

	Accuracy		Corrected area	
	Initial	*Gain*	*Mean*	*Max*
LinkNet34	86.4	0.27	130	372
Unet-eff	88.6	0.13	345	993
DeepLabv3+	86.4	0.4	281	987
TransFuseLinkNet	86.8	0.36	404	1010
TransFuseUnet-eff	88.5	0.1	341	998

Table 4.2 – Accuracy gain and spatial impact of an annotation on the ISPRS Potsdam dataset with 1024×1024 patches.

We performed a similar experiment with patch sizes of 1024x1024 instead of 512x512. As we can observe on Table 4.2, we obtain similar results and, in particular, Unet-eff and DeepLab architectures are still not limited by their receptive field (the distances of the corrected pixels from the annotations grow linearly with the pach size). It should be noted that the IoU gains are necessarily lower than with smaller patches since we are working with as many annotations on a larger surface. However, LinkNet does not correct pixels further than when dealing with 512×512 patches. This is fixed by TransFuseLinkNet, which triples its sphere of influence and leads to slightly better improvements.

	IoU		**Corrected area**	
	Initial	*Gain*	*Mean*	*Max*
LinkNet34	$85.2^{+- 0.01}$	$0.86^{+- 0.3}$	132	252
Unet-eff	$88.9^{+- 0.01}$	$0.47^{+- 0.25}$	203	374
DeepLabv3+	$87.3^{+- 0.01}$	$0.75^{+- 0.3}$	207	397
TransFuseLinkNet	$89.4^{+- 0.01}$	$0.84^{+- 0.3}$	231	415
TransFuseUnet-eff	$91.8^{+- 0.01}$	$0.3^{+- 0.2}$	208	365

Table 4.3 – IoU gain and spatial impact of an annotation on the AIRS dataset with 512×512 patches.

The results on the AIRS dataset, displayed on Table 4.3, show an important gap between the initial accuracy of fully convolutional architectures and the one from their transformer counterparts, but with modifications within the same ranges of magnitude.Regarding the propagation of the annotations, LinkNet is again behind the others but this lack of long range ability is addressed by the transformer component. However, for Unet-eff, the convolutional architecture is already able to propagate efficiently the information of the annotation and the transformer component only helps to improve the initial prediction.

To conclude, we have mainly explored the addition of a Transformer branch in parallel of a convoluted one in CNNs within our DISIR framework. It appears that small architectures with a low receptive field propagate annotations better with this transform component. However, the interest of small architectures being precisely their computational lightness, the addition of a transformer component actually appears counterproductive. On the other side, larger convolutional architectures do not need an additional transformer branch, at least for patches up to 1024x1024 since the metric gains are similar. It might be more beneficial with even larger patches but we are currently by our computing resources for such experiments. The computational cost of transformers does not allow us to currently use them efficiently in our interactive context.

All of these mitigated results has discouraged us to pursue this lead any further. The difficulty to increase the receptive field with a light model (thus adapted to

interactivity) invites to consider retraining despite the possible instabilities, since it also allows to propagate information between the different patches within the entire image.

IV.B . DISCA : Deep Image Segmentation with Continual Adaptation

IV.B.1 . Motivation & contribution

In Chapter III, we have proposed a methodology to refine segmentation maps with user annotations without any retraining. Indeed, this only modifies the inputs of the algorithms. This has the nice property of being extremely robust and ensures that the modifications are localized around the annotations. While this can be a desired behavior, it can be also useful to propagate the information carried by the annotations all across the images. With this in mind, we have proposed to add self-attention layers in the neural network architectures to be free from the spatial limitation inherent to convolution receptive fields. However, we have obtained mitigated results and seen that we are then limited by computing power and by the patch size processed.

Hence, aiming to address both the intra-patch propagation and the inter-patch propagation issues, we now propose to fine-tune the neural network on-the-fly using the user annotations as a sparse ground-truth. However, a standard fine-tuning has two major flaws. First, the network would lose its ability to locally improve its results like with DISIR. Second, it could over-fit on this extremely small training set. To address the former issue, we simply combine the fine-tuning process with our DISIR mechanism. For the latter, we also introduce a customized regularization to enforce stable and consistent predictions. Through experiments, we show the efficiency of this methodology, despite its relative slowness due to the retraining component. It seems especially suitable for the correction of relatively large errors and segmentation maps with many mistakes. Specifically, we show its potential with different applications in domain adaptation scenarios. Most of the works presented in this chapter from now on have been published in **Interactive Learning for Semantic Segmentation in Earth Observation** Lenczner G., Chan-Hon-Tong A., Luminari N., Le Saux B. & Le Besnerais G., ECML-PKDD MACLEAN Workshop 2020 and the associated code is available on this GitHub repository : https://github.com/delair-ai/DISCA.

IV.B.2 . Methodology

Since DISIR only modifies the network's inputs and not its parameters, the information provided by the annotations does not improve the predictions globally in the image. Inspired by What's The Point (WTP) [Bearman et al., 2016], we propose with DISCA to bypass this locality constraint by retraining the network with a few back-propagation cycles per annotation. The general methodology is illustrated in Figure 4.1.

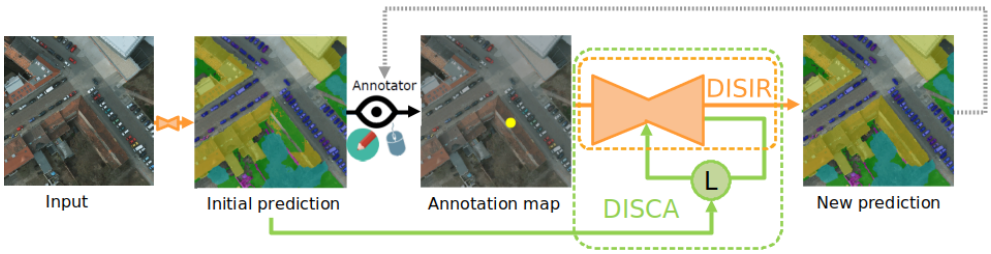


Figure 4.1 – Visual representation of DISCA. The framework starts with an initial prediction that the annotator can annotate with new points (e.g. to fix errors). Best viewed in color.

Hence, we use the annotations as a sparse ground-truth to interactively retrain the network using a cross entropy loss on these annotated pixels. It is noteworthy that DISCA builds on DISIR, which means that the annotations also modify the network inputs. We note $\mathbf{f}$ to represent the neural network parameterized by θ and $\mathbf{x}$ its inputs. As only a few pixels are annotated among the millions that usually compose a remote sensing image, the ground-truth maps resulting from the interactions are extremely sparse. In order to deal with this problem and avoid over-fitting, we follow ideas from [Kontogianni et al., 2020, Caye Daudt et al., 2019] by using the initial prediction $\mathbf{p_0} = \mathbf{f}(\mathbf{x}, \theta_0)$ for regularization. Precisely we add a $L1$-loss term using the original prediction as reference in order to prevent the model from making a prediction too different from the initial one. Therefore, our loss during the interactive learning process is defined as follows :

$$\mathcal{L}(\mathbf{x}, \mathbf{c}, \mathbf{p_0}; \theta) = \frac{\mathbf{1}_{[\mathbf{c}\neq -\mathbf{1}]}}{\|\mathbf{1}_{[\mathbf{c}\neq -\mathbf{1}]}\|_1} \left\{ -\sum_{i=1}^{N} \mathbf{c}_i \log\left(\mathbf{f}_i(\mathbf{x}; \theta)\right) \right\} + \lambda \|\mathbf{f}(\mathbf{x}; \theta) - \mathbf{p_0}\|_1 \tag{4.1}$$

where $\mathbf{1}$ represents the indicator function and $\mathbf{c}$ the sparse annotated pixels. In details, $\mathbf{c}$ takes its values in {-1, 0, 1}. For the pixels annotated as belonging to class i, $\mathbf{c}_i = 1$ and $\mathbf{c}_j = 0$ for all $j \neq i$. For the unannotated pixels, $\mathbf{c}_i = -1$ for all i in $\{1, \ldots, N\}$. Finally, the positive parameter λ balances the influence of user annotations with respect to the recall towards the initial prediction. Its tuning will be considered in Section IV.B.3.b.

During the interactive training phase, the DISIR mechanism is randomly disabled : the annotations are then removed from the inputs and only used as labels. This avoids over-fitting on the annotation channels.

To summarize, the DISCA module leverages three mechanisms :

1. DISIR : The annotations modify the neural network inputs both during training and inference.
2. WTP : It uses the annotations as a ground-truth for interactive retraining.
3. Finally, a regularization term based on initial predictions is crucial to com-

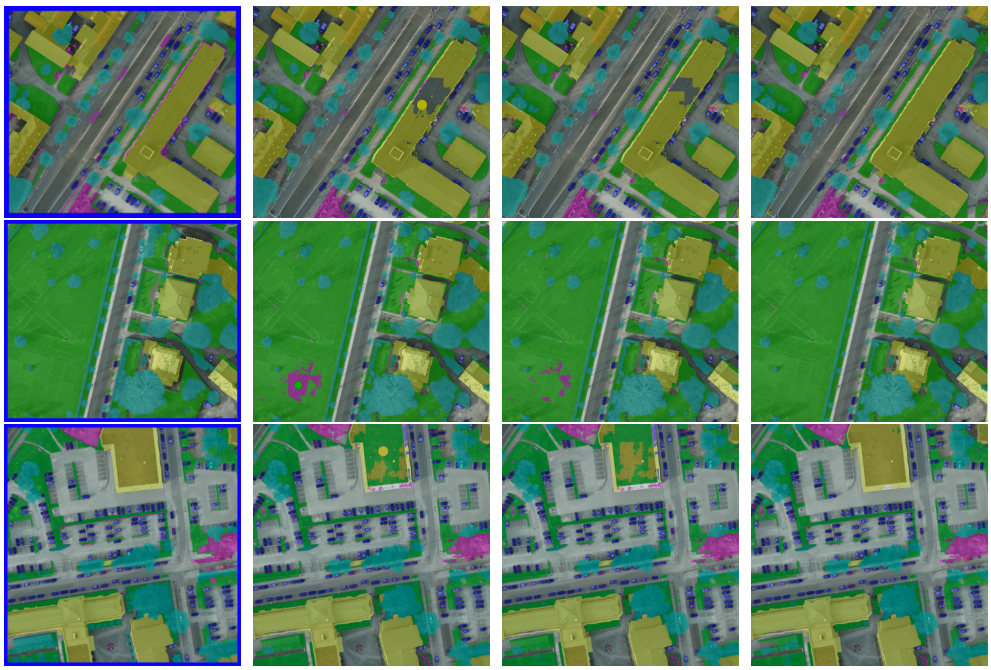
plement the cross-entropy loss during retraining.

The two last mechanisms enable the continual learning potential of DISCA and avoid catastrophic forgetting.

IV.B.3 . Experiments

To evaluate the refinement performances on the validation sets, we sample 10 annotations from the ground-truth maps in the largest wrongly predicted areas, adapt the networks in an image-wise fashion and measure the IoU evolution. During the interactive learning phase, we optimize the weights using 10 stochastic gradient descent passes with a learning rate of $2e^{-7}$ and minimize the loss defined in Eq. 4.1.

IV.B.3.a Approach assessment



(a) Ground-truth (b) Initial prediction with one annotation (c) DISIR (d) DISCA

Figure 4.2 – Visual results on the ISPRS dataset with single annotations. Best viewed in color.

As shown in Table 4.4, DISCA successfully enhances the initial segmentation maps to reach a higher IoU. Indeed, we observe an average improvement of 2.5% IoU with ten annotation samples. Besides, it also beats DISIR performances on the three datasets.

DISCA efficiently allows the user to make corrections at the image scale : on Figure 4.2, single annotations enable DISCA to provide a corrected segmentation of

	Mean IoU			Δ IoU	
	Initial	*DISIR*	*DISCA*	*DISIR*	*DISCA*
ISPRS	70.7	71.7	**72.4**	1	**1.7**
INRIA	85.4	86.4	**86.5**	1	**1.1**
AIRS	88	89.8	**90.2**	1.8	**2.2**

Table 4.4 – Performances in terms of mean IoU before and after the interactive processes with 10 annotations per image.

	Initial	*DISIR*	*DISCA*
time (s)	0.01	**0.01**	0.11

Table 4.5 – Mean inference time on a 512×512 patch

the scenes while they are not enough for DISIR to deliver a similar result. However, this has to be moderated by the inference time of the two algorithms. Indeed, as shown in Table 4.5, DISCA is more than $10\times$ slower than DISIR due to its retraining component.

IV.B.3.b Ablation study and comparison with state-of-the-art

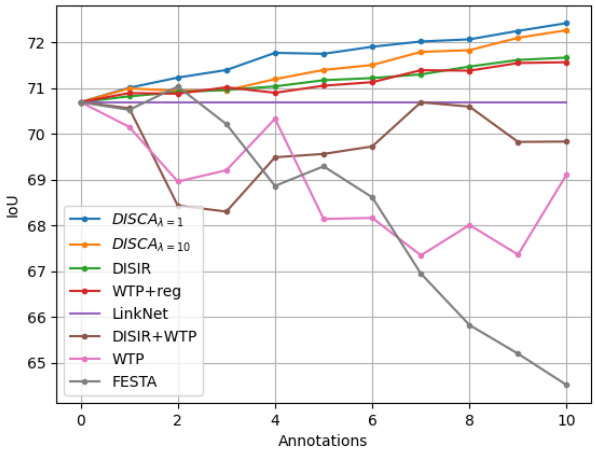


Figure 4.3 – Ablation study on ISPRS dataset.

To understand the influence of the aspects of the DISCA algorithm, we analyze separately its different components. DISIR adds annotations input channels. WTP [Bearman et al., 2016] retrains the model based on a few annotations. DISCA sums up DISIR and WTP with regularization with respect to the initial prediction.

We also test DISIR combined with WTP, and WTP combined with regularization. To study the importance of the regularization parameter λ, we test two values DISCA$_{\lambda=1}$ and DISCA$_{\lambda=10}$. Finally, we also compare our models to FESTA [Hua et al., 2021] which trains a neural network on point annotations (as WTP) with a different regularization that will be detailed in Chapter VI. As shown on Figure 4.3, DISIR and WTP+reg obtain IoU gains around 1% for 10 clicks and are beaten by the various flavors of DISCA which almost doubles the gain. This means that the interactive retraining process could be effectively applied to any classically trained neural network but needs to be combined with the DISIR process to fully exploit the annotations. Moreover, we observe that the regularization is extremely important in DISCA as its absence leads to worse results (DISIR+WTP curve) than the initial ones (LinkNet curve). A too high λ also decreases the benefits brought by DISCA because it then prevents the algorithm to optimally exploit the annotations. Finally, in this framework of incremental learning, WTP [Bearman et al., 2016] and FESTA [Hua et al., 2021] also lead to worse results than the initial ones, as emphasized in Table 4.6. These methods were originally designed to train the neural networks from scratch on point annotations. Hence, it explains why they are not optimal in such refinement scenario since they take into account different constraints.

WTP [Bearman et al., 2016]	69.1
FESTA [Hua et al., 2021]	64.5
DISIR (ours)	71.6
DISCA (ours)	**72.4**

Table 4.6 – Comparison between different regularization on ISPRS Potsdam dataset after 10 annotations

We also compare our approaches with the ALCD method [Baetens et al., 2019], also deployed in the field of remote sensing for cloud segmentation in low resolution (60 m/pixel) images. To adapt it to our use-case, we run ALCD in a fine-tuning setting on the ISPRS dataset. In practice, we initially pre-train the ALCD random forest on $100\,000$ samples per image from the training set, and then adapt the classifier with the same number of annotations as DISIR and DISCA. However, it leads to very poor performances, both before (30% IoU) and after fine-tuning (30.5% IoU), compared to DISIR/DISCA results presented previously. While the absolute results might be due to differences of peculiar implementations of random forest and neural network, the ALCD gain is only $+0.5\%$, which is 2 times less than DISIR and 3 times less than DISCA for the same amount of annotations.

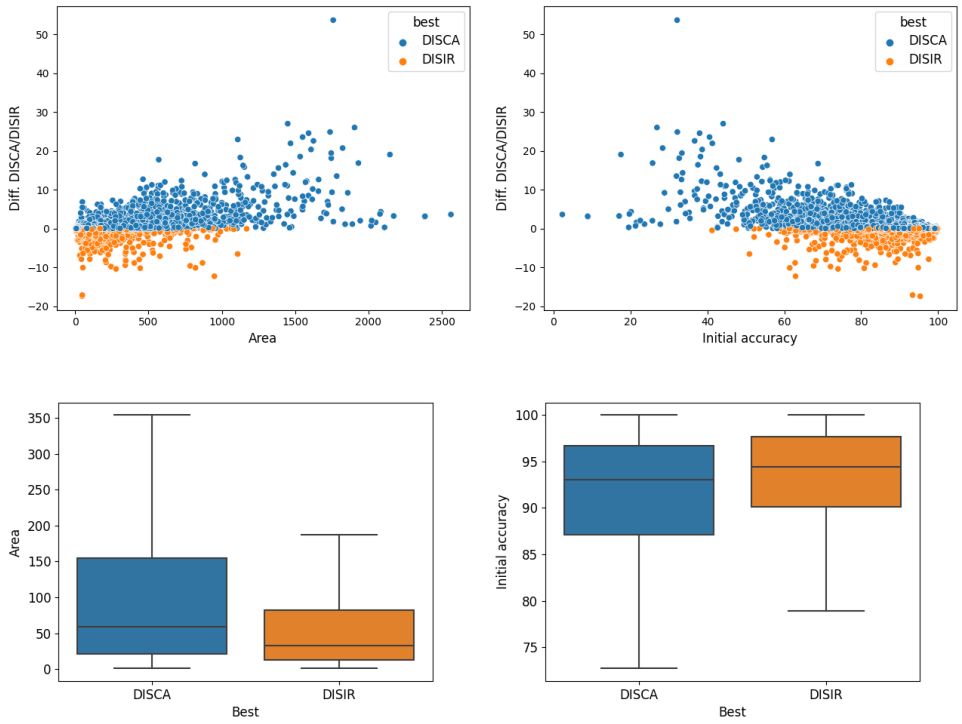


Figure 4.4 – Comparison of DISIR and DISCA (IoU) with respect to the spatial size of the corrected mistake and the initial accuracy. Legend "best" designates the best method for the given sample.

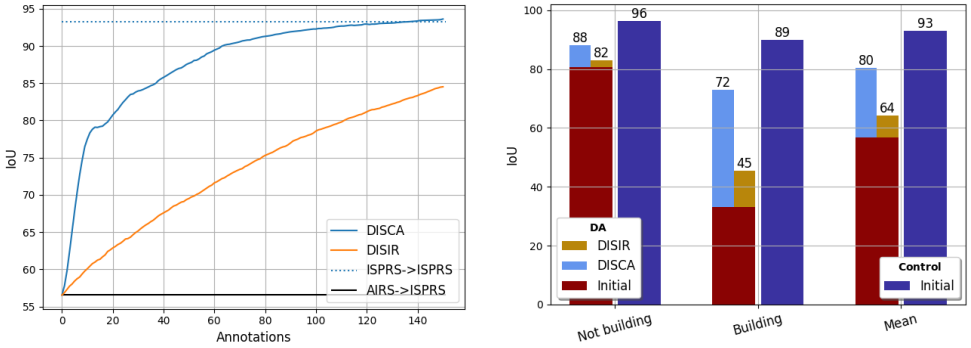
IV.B.3.c Influence of initial segmentation conditions, or : when to choose DISIR or DISCA ?

To better apprehend the difference between the two methods, we sample $10\,000$ 512×512 patches from each dataset. Then, given one annotation, we compare the difference between DISIR and DISCA based on two parameters : the spatial size of the corrected mistake and the initial accuracy of the network on the patch. Precisely, the spatial size of the corrected mistake is the size of the error polygon in which the annotation is sampled. Similarly to the initial accuracy, it is obtained with a comparison between the initial predicted map and the ground-truth map. It is intuitively obvious that both DISIR and DISCA are correlated to these parameters since, if the mistake to correct is small, the overall IoU gain will be smaller than with a larger mistake to correct. However, we think that this comparison can bring valuable insights to choose the appropriate method depending on the situation.

Figure 4.4 compares the two methods with respect to these two criteria. First, both methods seem to work well and can outperform the other one when the mistake area is small and the initial performance is high. We thus recommend using DISIR in these situations. Indeed, the locality of DISIR is no longer a constraint since the error is strongly spatially contained and the relatively long retraining time

inherent to DISCA makes it less suitable here. Second, when the initial accuracy is low or the area to correct large, DISCA now clearly tends to perform better than DISIR, and we thus believe that it should be favored in these situations. Indeed, its spatial globality resulting from its retraining can be fully expressed to correct large mistakes. This outcome shows that DISCA is more relevant to correct deeply flawed segmentation maps than DISIR.

IV.B.3.d Domain adaptation



(a) Mean IoU w.r.t the number of annotations.

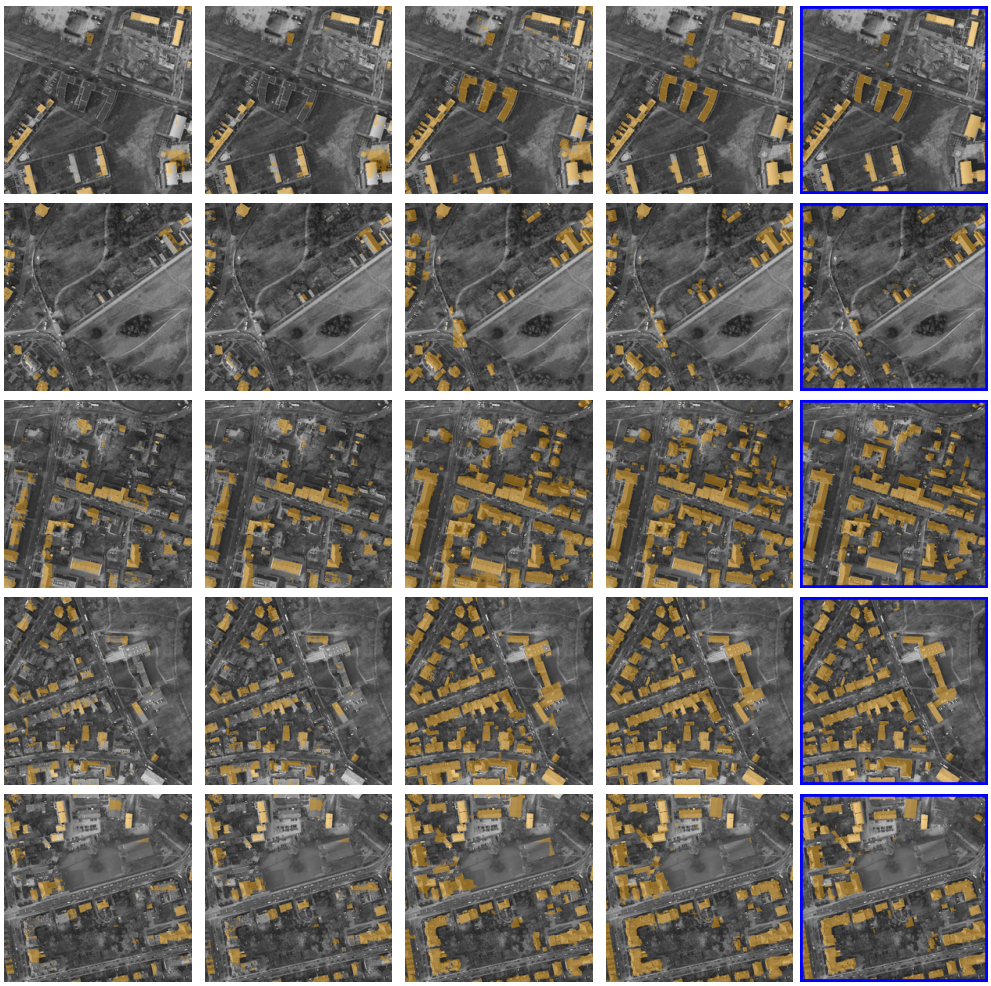
(b) IoU after 20 annotations

Figure 4.5 – Mean IoU of DISIR and DISCA for domain adaptation (AIRS→ISPRS).

To push DISCA a step further on flawed segmentation maps, we evaluate it in a domain adaptation scenario.

The objective in this domain adaptation use-case is to detect buildings on the 8 images of the ISPRS validation set. To this purpose, we compare a neural network trained on AIRS under DISIR and DISCA settings to a control one trained on the ISPRS training set. The ISPRS images are down-sampled using bi-linear interpolation to the AIRS resolution. The neural network's weights are reinitialized between each image. Figure 4.5 shows that a network weakly supervised with DISCA beats DISIR by a large margin in this scenario. Besides, it can quickly reach high performances (more than 80% IoU within 20 annotations) and even outperform a fully supervised one with a sufficient amount of annotations. This is visually confirmed on Figure 4.6. Indeed, 10 annotations enable the network to well understand the new domain images and thus propose decent segmentation maps. More annotations correct most of the remaining mistakes. In particular, the network is able to adapt to buildings with peculiar roofs or of uncommon size with respect to the AIRS dataset.

This shows that DISCA can be relevant to enhance mitigated initial results when dealing with domain adaptation and applying a pre-trained model to new geographical areas, which is a standard use-case of many applications.

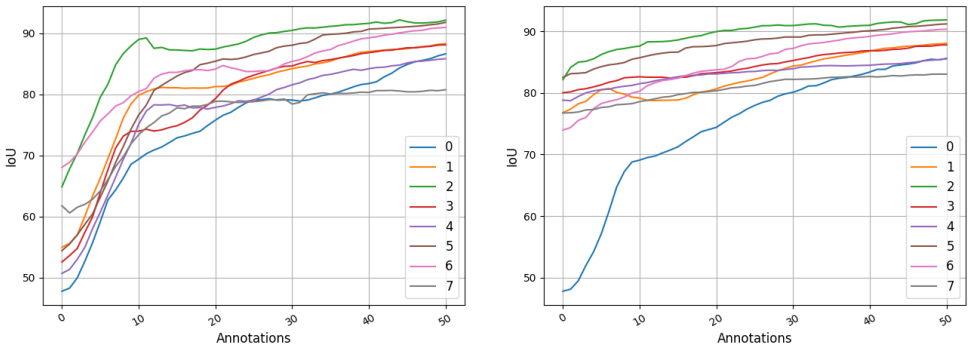


(a) Initial output (b) 1 annotation (c) 10 annotations (d) 50 annotations (e) Ground-truth

Figure 4.6 – Domain adaptation (AIRS → ISPRS) visual examples

IV.B.3.e Sequential learning

Finally, we analyze the generalization of DISCA through a sequence of images in the same domain adaptation scenario. This means that we do not reinitialize the neural network weights between each image. We refer to this set-up as sequential learning, and we learn two insights from it on Figure 4.7. First, DISCA does not suffer from catastrophic forgetting here as the algorithm does not diverge even on the last seen images. Second, sequential learning greatly improves the initial performances directly after the first image. Indeed, the initial IoU then approximately increases of 20%. However, after few annotations, the sequential learning benefits vanish and the performances become similar to the non-sequential set-up.



(a) Weights reinitialized between each image.

(b) Weights updated between each image.

Figure 4.7 – Sequential learning study with DISCA in a transfer scenario. The legend corresponds to the order in which the algorithm processes the images.

IV.B.4 . Conclusion

We have first proposed in this chapter to enlarge the neural network receptive fields for a better intra-patch propagation of the annotations, relying on Transformers and on the self-attention mechanism. However, this resulted in mitigated results and cumbersome architectures, which are not entirely suitable for interactive approaches. Moreover, this did not allow to address the inter-patch propagation of the annotations.

Hence, to address both the intra and inter-patch propagation challenges simultaneously, we have also proposed an interactive multi-class segmentation framework for aerial images, which complements the one proposed in Chapter III. The refinement scheme is based on the retraining of the neural network using the annotations provided by the user as a sparse ground-truth. This also leverages a $L1$-regularization which prevents over-fitting by using the initial prediction as an anchor. We have shown through experiments its efficiency, specifically when the algorithm proposes a mitigated initial segmentation map like in domain adaptation scenarios.

These results have great potential in industrial applications, given the ease of adaptation of the network and the open-source release of our code. In particular, beyond the applications within Alteia and ONERA, DISCA is now also used and applied by another company in the context of 3D buildings reconstruction from semantic segmentation maps [Dumas et al., 2022]. Starting from a neural network trained on a small part of the world and deployed on a larger scale, their application framework is thus close to the one established in our experiments. We hope that our work will be involved in new industrial and scientific applications in the future.

We have thus shown that DISCA effectively addresses the spatial limitation inherent to DISIR, and that it can even propagate information between images. However, it is necessary to be lucid and to analyze the various limits of the proposed approach.

First, the processing time is increased due to the retraining component, which

is an undesirable behavior in an interactive framework. This makes the use of DISIR completely viable depending on the situation and the two approaches are complementary. Processing larger patches (or even the entire image), combined with large architectures capable of propagating the information well, could combine the speed of DISIR and the propagation of DISCA. However, we have seen with our Transformers experiments that we are not there yet. Not excluding possible errors in our implementations, we would still need significant computing power (beyond our currently available resources) to be able to use Transformers on large patches and exploit these architectures fully.

Second, DISCA brings three additional hyper-parameters that need to be tuned : the learning rate, the number of learning steps and the regularization weight. An exhaustive research on these different parameters could bring additional valuable insights. Moreover, it could also be interesting to deepen the work on the regularization term, possibly by going into the field of self-supervision. Indeed, while the current regularization term relying on the initial prediction guarantees some stability, it also possibly prevents some desired changes. Specifically, it might be sub-optimal for domain adaptation cases where many changes usually need be made.

Third, although errors can be corrected with DISIR and DISCA, it can be tedious for an annotator to find where the errors are. This is especially true in remote sensing where the images can be very large. Therefore, moving closer to active learning, we propose to remedy this in the next chapter by determining the uncertainty of the algorithm to guide the user to likely erroneous areas.

Finally, in the current configuration, DISCA only enables to fine-tune the network on the same segmentation classes. This strongly constrains the approach but let the neural network keep its initial structure. Indeed, a change in the segmentation classes to be predicted necessarily implies to modify the last layer of a neural network. We will address this issue in Chapter VII.

Chapter V - Guiding the interactions

Contents

V.A . Motivation & contributions

So far in this thesis, we have mainly studied the first axis defined in the Introduction : *How to interact with neural networks after learning for semantic segmentation ?* and have presented DISIR and DISCA algorithms that can both be used to refine segmentation proposals made by the network. However, remote sensing images can be extremely large and it can then be tedious for an operator to review them entirely. To address this issue, we now propose a methodological improvement relying on active learning to swiftly guide the agent towards queries representing the most meaningful areas of the images to annotate. We thus focus here on our second research axis : *How to choose which data to annotate ?*

As explained in Chapter II, Active Learning [Settles, 2009] (AL) searches in pools of unlabeled data for examples which are the more able to lead the model to a better classification. These examples, defined as *queries*, are then labeled by the user and incorporated in the training. This thus aims to find the optimal training dataset for

the algorithm. To this purpose, active learning methods define *acquisition functions* to estimate either the model uncertainty associated to the samples [Gal et al., 2017] or their representativeness of the dataset [Sener and Savarese, 2018]. In our case, we do not look for an optimal training dataset but to correct errors on segmentation maps as efficiently as possible. Hence, while the representativeness of the dataset is not relevant here, we will see if we can use the uncertainty measures used in active learning for our use case like in [Lewis and Catlett, 1994].

The methodology proposed in this Chapter is meant to be used with both DISIR and DISCA. The additional guidance relies on different uncertainty measures that we apply with respect to our framework. These measures can be simple-yet-effective such as entropy [Shannon, 1948] or come from the state-of-the-art literature such as ODIN [Liang et al., 2018] or ConfidNet [Corbière et al., 2019]. The works presented in this chapter have been published in **Weakly-supervised continual learning for class-incremental segmentation** Lenczner G., Chan-Hon-Tong A., Le Saux B., Luminari N., & Le Besnerais G., IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing. The associated code is available on this GitHub repository : https://github.com/alteia-ai/DIAL.

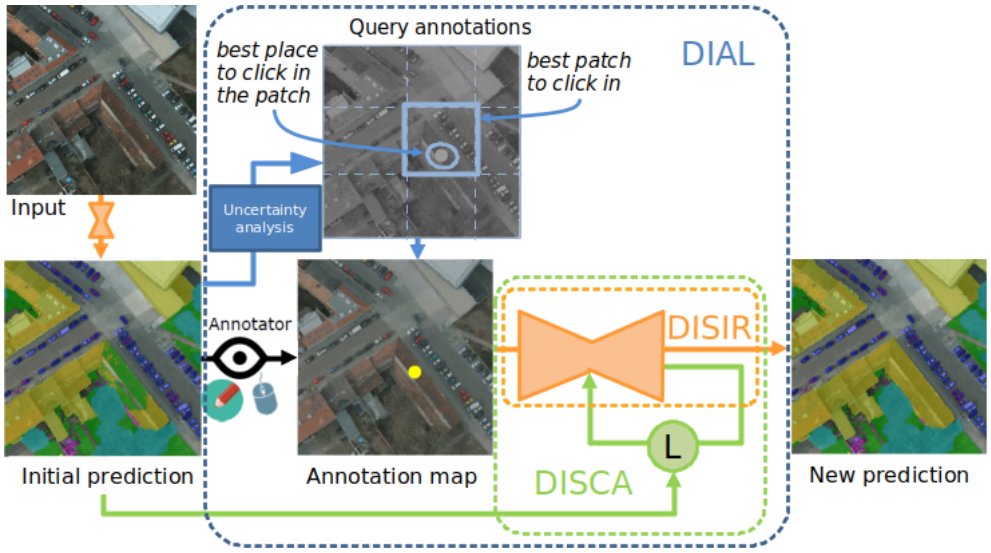


Figure 5.1 – Visual representation of DIAL encompassing Deep Image Segmentation with Continual Adaptation (DISCA) and Active Learning. The framework starts with an initial prediction that the annotator can annotate with new points (e.g. to fix errors). Three algorithmic mechanisms cooperate to improve the segmentation map : DISIR processes jointly image and annotations without retraining, DISCA additionally retrains the model for better adaptation, and DIAL also proposes most informative patches to speed up the interactions. Best viewed in color.

V.B . DIAL : Deep Interactive and Active Learning

In order to guide the agent towards relevant queries, we compare different state-of-the-art acquisition functions that estimate the algorithm uncertainty to find the most suitable for our interactive set-ups. As it builds upon DISIR and DISCA algorithms, we name the complete framework DIAL : Deep Interactive and Active Learning. The framework is illustrated in Figure 5.1.

Uncertainty quantification in computer vision is often classified into two categories [Kendall and Gal, 2017]. First, aleatoric uncertainty captures noise inherent in the observations, which typically cannot be reduced. This can for example come from the sensor used to collect the data. Second, epistemic uncertainty models the uncertainty in the model. This is usually due to a lack of knowledge of the algorithm and it can then potentially be reduced by using more training data. We show on Figure 5.2 examples of these two types of uncertainty in the ISPRS Potdam dataset. Since each of these two types of uncertainty can lead to classification errors that potentially need to be corrected, we currently do not distinguish between them within our interactive framework.

(a) Noise from photogrammetric reconstruction

(b) Only "*building*" with "*cars*" on it in the dataset.

Figure 5.2 – On the left, the photogrammetric noise leads to aleatoric uncertainty. On the right, the lack of parking lots in the dataset leads to epistemic uncertainty. More similar data would improve the recognition of the semantic "*car*" surrounded by "*building*".

V.B.1 . Problem formulation

To formulate the problem, we note $\mathbf{f}$ to represent the neural network parameterized by θ, $\mathbf{x}$ an input image, $\mathbf{y}$ its associated label map, $\mathbf{a}$ the user annotations and $\mathbf{g}$ the annotation encoding function. Our goal is then to find the optimal annotations $\mathbf{a}^\star$ minimizing the following problem :

$$\begin{aligned} \mathbf{a}^{\star} &= \underset{\mathbf{a}}{\operatorname{argmin}} \sum_{j \in I} \left(1 - \delta_{\mathbf{y}^j}^{u^j}\right) \\ \text{with } u^j &= \underset{c \in \{1,\dots,N\}}{\operatorname{argmax}} \mathbf{f}_{\theta,c}^{j}\left(\mathbf{x} \oplus \mathbf{g}(\mathbf{a}, \mathbf{x}, \mathbf{f}_{\theta})\right) \end{aligned} \tag{5.1}$$

where $\oplus$ represents the concatenation operation, δ the Kronecker operator, N the cardinal of the label space and I the pixels set. The problem values range from 0 when all pixels are well classified to $card(I)$ when all pixels are misclassified.

V.B.2 . Query strategies

We identify two possible query strategies to benefit from DIAL on a given image :

1. The **patch-based strategy.** The image is divided into a grid of N patches. The patches are annotated consecutively but the order in which they are annotated depends on the uncertainty measure. The clicks are then sampled in the largest erroneous areas using a comparison between the ground-truth and the prediction.
2. The **pixel-based query strategy.** The image is divided into a grid of N patches and these patches are given to the agent always in the same order for reproducibility reasons. The uncertainty is used to guide the agent where to annotate within the patch.

The pixel-based query strategy is probably the most intuitive one. It allows to figure out whether the uncertainty measurements can help to spot errors at a pixel level. Moreover, it can help to determine whether, among misclassified pixels, those with high uncertainty become more informative annotations than the ones with low uncertainty. However, this approach is possibly unstable and too much localized : it could be more useful to think in areas to correct instead of pixels.

The patch-based strategy addresses these by smoothing out the uncertainty. It allows to study the order of the patches to annotate. For instance, in Figure 5.1, the middle patch would be selected first and the agent would correct it, and the process would be repeated for the other patches, always chosen according to the uncertainty. Moreover, because it has only an intra-patch impact, DISIR ends up at the same point for an image in this setting whatever the patches order but not at the same speed. This allows to consistently compare the different uncertainty acquisition functions according to our interactive framework.

Therefore, these two strategies are complementary and can be linked up in a multi-scale scheme. Indeed, the first one is global with patch-based queries as it looks for the optimal patches to annotate while the second one is local with pixel-based queries since it guides the agent inside the patch.

V.B.3 . Acquisition functions

We now present the different acquisition functions that we compare to guide the agent. In relation to the categories defined in Chapter II, two of these functions rely directly on the softmax probabilities of the algorithm, one on an ensemble method and one on an auxiliary model. Other acquisition functions could also be used [Ružička et al., 2020, Besnier et al., 2021, Lee et al., 2018].

V.B.3.a Entropy

We compute the entropy per pixel at the softmax output : $\mathcal{U} = -\sum_c y_c \times \log(f_c(x;\theta))$. As showed by [Hendrycks and Gimpel, 2017], even though the softmax probabilities of a neural network are poorly calibrated, they can still provide a strong baseline to guide the user.

V.B.3.b MC Dropout

MC Dropout [Gal and Ghahramani, 2016] introduces stochasticity in the prediction by enabling dropout regularization at inference time. This allows to obtain an implicit model ensembling. In practice, we add dropout layers in the neural network architecture and then make multiple forward passes through the network to create as many softmax vectors. We then compute the variance of these predictions to measure their disagreement and use it as the uncertainty measure.

V.B.3.c ConfidNet

As proposed by [Corbière et al., 2019], we train a small auxiliary network to learn to estimate the confidence value of the downstream network using its last layers as inputs. It is constituted of one transposed convolutional layer and four 3×3 convolutional layers of respectively 32, 120, 64, 32 and 1 output layers. A final sigmoid layer provides the confidence score.

V.B.3.d ODIN

Following [Liang et al., 2018] which primarily developed this method for outlier detection, we slightly disturb the image input with an adversarial-like attack aiming to enforce the predicted probabilities of the softmax output towards the predicted classes and add a temperature term in the softmax layer. Then, the adversarial examples are feed-forwarded into the network and we use the softmax output maximum class probability as a confidence measure. Formally, we disturb the input with the following perturbation $\mathbf{x} = \mathbf{x} + \varepsilon \Delta_{\mathbf{x}} \mathcal{L}(f_\theta(x), \hat{y})$ where $\mathcal{L}$ represents the cross-entropy loss, $f_\theta(x)$ the predicted probabilities from the softmax output and $\hat{y}$ the predicted class.

V.B.3.e Computational cost

These approaches have different inference costs inherent to their underlying structure. Indeed, entropy is virtually cost-free since it computes a simple operation directly on the neural network output. On the contrary, MC Dropout is particularly expensive since it requires computing multiple predictions. Despite the extra prediction, ConfidNet is only slightly more expensive than entropy thanks to the small size of the auxiliary network. Finally, ODIN falls between ConfidNet and MC Dropout due to the creation and inference of the adversarial sample.

V.C . Experiments

V.C.1 . Experimental set-up & hyper-parameters

To automatically evaluate this active learning component, we split the test images into 512×512 patches, sample one annotation per patch and then make a new prediction on this patch using DISIR and DISCA.

For ODIN, we set the perturbation parameter ε to $\frac{1}{255}$ and the temperature term to 100.

For MC Dropout, we add a dropout layer between each encoder and decoder block of our architecture, set the dropout rate to 0.1 and compute the variance over 5 different inferences.

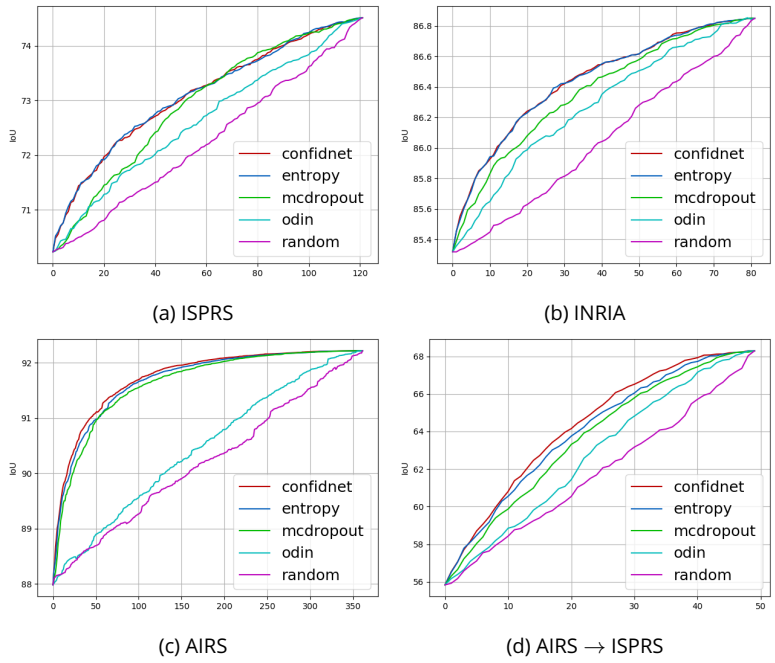
The ConfidNet auxiliary network is trained for 10 epochs per dataset with Adam optimizer.

V.C.2 . Patch-based query strategy

With the patch-based query strategy, we study whether the annotation order can be optimized. The annotations are generated inside the spatially-largest mistakes of the patches. We compute the uncertainty globally in the images and then compute an uncertainty score per patch by averaging the uncertainty across all the pixels of the patch. We compare the uncertainty-ordered sequences to a randomly-drawn one that constitutes the baseline.

	Initial	**Rand. patches**		**AL patches**	
	LinkNet	*DISIR*	*DISCA*	*DISIR*	*DISCA*
ISPRS	70.7	71.8	71.3	73.1	**73.3**
INRIA	85.4	86.3	86.2	**86.6**	86.4
AIRS	88	88.7	89.4	91.1	**92**

Table 5.1 – Mean IoU after 50 annotated patches with random and active learning (entropy) orders. For 50 patches on Figures 5.3& 5.4, one recovers results from this Table.



(a) ISPRS (b) INRIA

(c) AIRS (d) AIRS → ISPRS

Figure 5.3 – IoU evolution with respect to the number of annotated patches with DISIR (one annot. per patch). This compares the different uncertainty measures to select the patch-to-annotate.

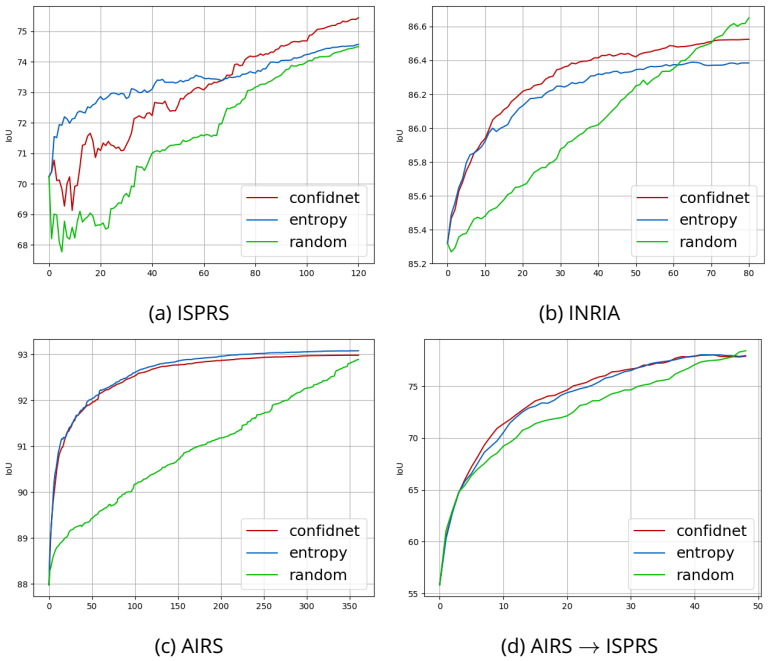
	Random	*Entropy*	*MC Dropout*	*ODIN*	*ConfidNet*
time (s)	9.9	**10.1**	22	15.8	12.6

Table 5.2 – Mean time for prediction with uncertainty computation on $6\,000 \times 6\,000$ images

V.C.2.a Active learning with DISIR

As we can see on Figure 5.3, the random order leads to an improvement linear with respect to the number of processed patches. All active learning schemes speed up the gain in performances by targeting the more uncertain areas. This is particularly noticeable on the AIRS dataset where 50 annotations are enough to reach 75% of the final improvement. This behavior is probably due to the dataset itself. Indeed, since it covers a lot of rural areas, many images only contain few buildings and the uncertainty measures then allow to quickly show the areas of interest to the user.

Regarding the different uncertainty measures, ODIN is consistently the worst one. It is only slightly better than the random order and, contrary to the other methods, its performance is almost linear on the AIRS dataset. This behavior might be explained by the method original purpose. Indeed, while the other methods aim to estimate the model uncertainty, ODIN aims to detect outliers. Though these



(a) ISPRS (b) INRIA

(c) AIRS (d) AIRS → ISPRS

Figure 5.4 – IoU evolution with respect to the number of annotated patches with DISCA (one annot. per patch). This compares the different uncertainty measures to select the patch-to-annotate.

tasks are related, it appears here that model errors are not due to this type of issue in the image area. Moreover, Table 5.2 shows that ODIN and MCDropout considerably slow the prediction process (resp. by factors 1.5 and 2) compared to entropy (factor 1) and ConfidNet (factor 1.2).

ConfidNet and entropy consistently obtain the best performances, with a slight advantage for the former on AIRS and the domain adaptation use-case. However, ConfidNet is also a bit slower and less flexible since it requires to train an additional network for each dataset. Eventually, entropy offers an excellent trade-off between high accuracy performances and fast computation, as it is only slightly slower than a random pick.

V.C.2.b Active learning with DISCA

Since DISCA slightly modifies the neural network parameters, we recompute the entire prediction and uncertainty after each processed patch. Since MC Dropout and ODIN already proved to be relatively slow and less performing with DISIR, we only compare entropy and ConfidNet in this set-up. As we can observe on Figure 5.4, results are more complex to interpret than with DISIR.

On ISPRS, the different methods are all a bit unstable, which is probably explained by the different improvements for the multiple classes of this dataset. However, both uncertainty methods still perform better than the random strategy

and the strategy relying on ConfidNet enables a gain up to 5% compared to 4% for the random one. On INRIA, both uncertainty strategies outperform the random one for the first 60 patches but end up being caught up, probably stuck in a local minimum and possibly due to the inconsistent ground-truth of the dataset. It is noteworthy that ConfidNet ends up outperforming entropy on these two datasets by a larger margin than with DISIR. On AIRS and in the domain adaptation situation, the behaviors are similar to the ones obtained with DISIR, with noticeably higher performances. Indeed, the gain are around 20% and 5% with DISCA while they were around 10% and 4% with DISIR respectively on the domain adaptation situation and the AIRS dataset.

Hence, these results confirm the benefits of a guidance towards relevant patches relying on uncertainty measures. *ConfidNet* is on average the best method to this aim. However, the faster, simpler and only slightly under-performing *entropy* is a very good alternative for successfully recognizing the most relevant areas to annotate.

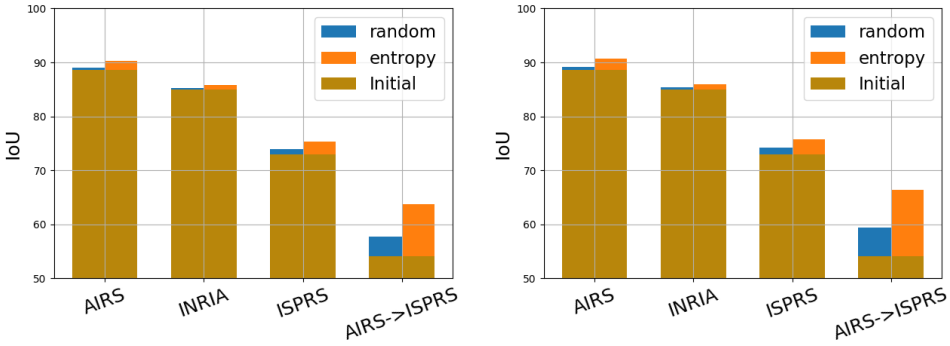
V.C.2.c Comparison to an upper-bound

	Initial	**Rand. patches** $\mathcal{O}(n_{\text{annots}} * d^2_{\text{patch}})$		**AL patches** $\mathcal{O}(n_{\text{annots}} * d^2_{\text{patch}})$		**Whole image** $\mathcal{O}(n_{\text{annots}} * d^2_{\text{image}})$	
	LinkNet	*DISIR*	*DISCA*	*DISIR*	*DISCA*	*DISIR*	*DISCA*
ISPRS	70.7	71	68.7	71.8	71.9	71.7	**72.4**
INRIA	85.4	85.5	85.5	86	86	86.4	**86.5**
AIRS	88	88.2	88.8	89.6	**90.7**	89.8	90.2

Table 5.3 – Performances in terms of mean IoU before and after the interactive processes with only 10 annotations per image, w.r.t. corresponding complexity. One patch corresponds to 4.4% of the whole image.

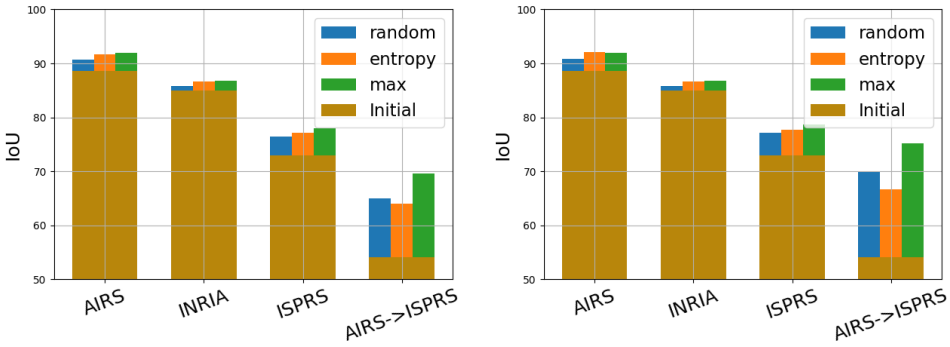
As shown in the previous section, an active learning patch order allows to better spot and correct mistakes than a random patch order, with both DISIR and DISCA. We compare it here to a theoretical upper bound of DISIR and DISCA : the agent generates each click at the center of the largest spatial error on the whole image, which would be optimal in terms of potential improvement but at the cost of a whole image search. As we can observe in Table 5.3, this leads to a 1.7% improvement with 10 annotations against a 1.5% improvement with the active learning strategy. As explained, this slight superiority is mitigated by the complexity to find the annotations. Indeed, in the whole image case, the agent has to browse through 3.6×10^7 pixels for each click in a $6\,000 \times 6\,000$ image (complexity : $\mathcal{O}(n_{\text{annots}} * d^2_{\text{image}})$) whilst, in the patch case, they have to browse through 2.6×10^5 pixels in a 512×512 patch (complexity : $\mathcal{O}(n_{\text{annots}} * d^2_{\text{patch}})$). Hence, it is 100 times more costly to find the annotation in an entire remote sensing image than in a patch. Therefore, the active learning strategy brings fluidity to the process, while leading to near-optimal performance.

V.C.3 . Pixel-based query strategy



(a) DISIR (b) DISCA

Figure 5.5 – Annotations sampled with uncertainty but without error knowledge



(a) DISIR (b) DISCA

Figure 5.6 – Annotations sampled using uncertainty and error knowledge

With the patch-based strategy, uncertainty successfully lead toward relevant patches to annotate but the annotations were then sampled in the selected patches without taking the uncertainty into account. With the pixel-based query strategy, we now test two more conjectures. First, we want to determine whether highly uncertain pixels **among the misclassified ones** can lead towards particularly meaningful annotations (i.e. to better corrections) with DISIR and DISCA. Second, we want to figure out whether the uncertainty measurements can help the agent to spot errors at a pixel level. We only rely on the entropy acquisition function here.

V.C.3.a Using uncertainty to look for optimal annotations

In order to look for optimal annotations, we make here the hypothesis that an agent always clicks on a wrongly segmented area and we compare the following annotations sampling strategies.

1. We sample the annotation randomly in the wrongly segmented area (*random*).
2. Like in the other experiments, we sample the annotation in the middle of the spatially-largest wrongly-segmented area (*max*).
3. We threshold the uncertainty map at the ninth quantile to keep only the highest uncertainty values. We then sample the annotation in the intersection of the wrongly-segmented area and this thresholded uncertainty map.

As shown on Figure 5.6, the uncertainty-based annotations lead to corrections of the same magnitude than the random ones on average. Moreover, these uncertainty-based annotations clearly don't provide more information to the model than the ones based on *max*. Indeed, the gains of *max* annotations with DISIR are around 6.4% IoU against respectively 4.5% and 4.7% for the random and uncertainty-based ones.

Therefore, this corroborates the correlation between the gain and the size of the corrected area previously exhibited and shows that uncertainty does not lead towards more meaningful annotations than the ones contained inside large mistakes.

V.C.3.b Uncertainty to spot mistakes

In order to evaluate if the uncertainty measures can help to spot mistakes at the pixel level, we compare annotations sampled randomly and on the basis of uncertainty measures *without* ground-truth prior knowledge. In other words, we do not coerce the annotations to be sampled in mistake areas.

Figure 5.5 shows that the uncertainty-based annotations lead to better improvements (3.7% IoU with DISIR, 4.5% with DISCA) than the random ones (1.3% IoU with DISIR, 1.9% with DISCA) on average. We can visually confirm these insights on Figure 5.7 where uncertainty measures tend to highlight wrongly predicted areas. Besides, the highlighted areas which are initially correctly predicted tend to be legitimately questionable such as object contours or road surfaces looking like buildings (third row).

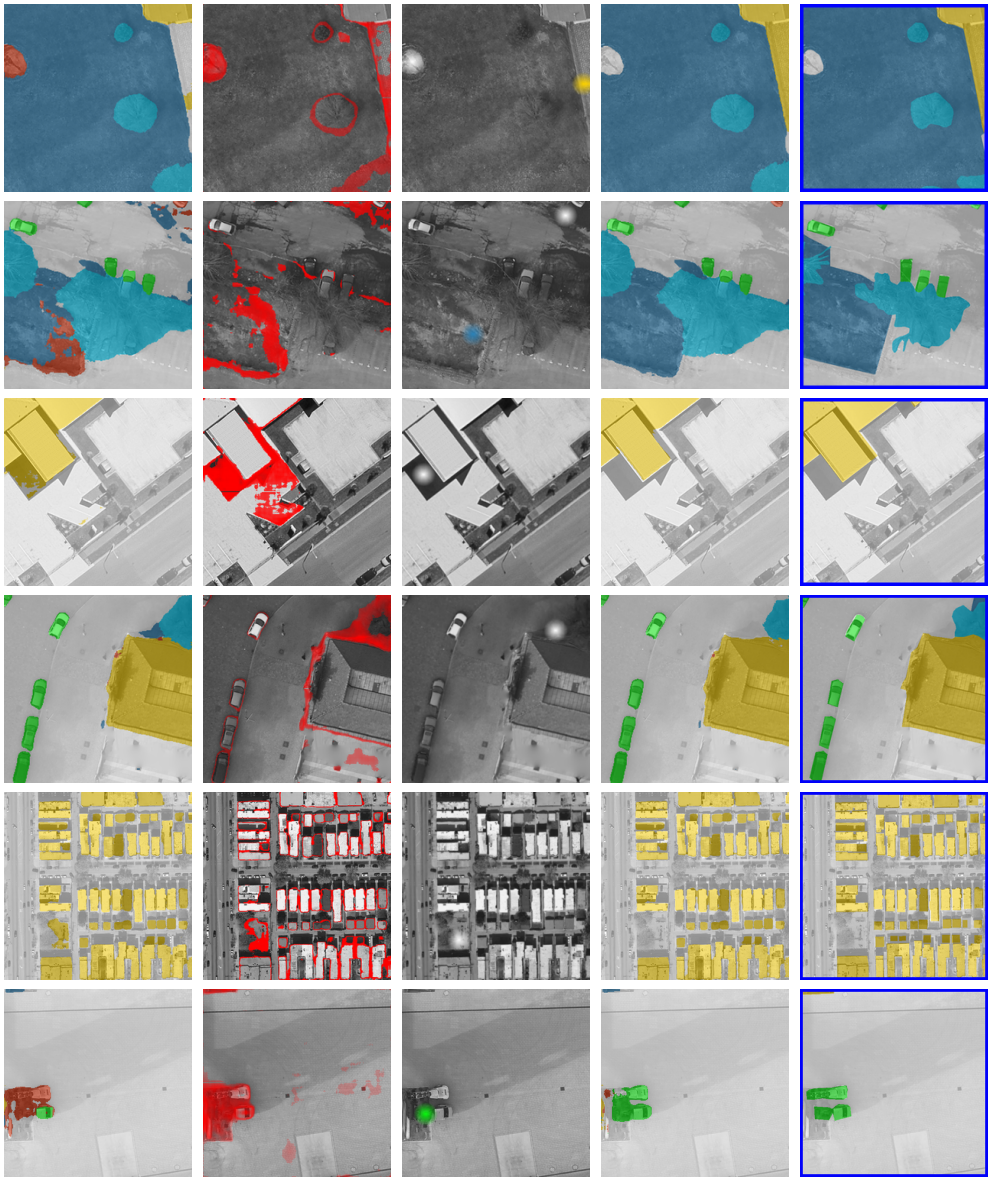
Therefore, even though uncertainty does not lead towards optimal annotations for DISIR or DISCA, it can be used as an additional help to detect mistakes at the pixel level.

V.D . Conclusion

We have shown that uncertainty measures are highly efficient to guide the agent actions toward insightful queries able to improve the classification performances. At patch level, they are always relevant to improve the choice of the areas to annotate. At pixel level, they can be used as a rough proxy for mistake detection over random, but they do not lead towards more insightful annotations, and the agent's ability to spot error regions is key to a truly improved model. Moreover, among the compared

acquisition functions, entropy offers the best trade-off for being simple-yet-effective while the recent ConfidNet measure leads (slightly) to the highest improvement if only accuracy is considered. If compared to random picking (i.e. interactive-only learning), active learning reaches faster high overall performances, and thus allows to reduce the number of interactions to match a given classification accuracy.

Therefore, this active learning-based methodology complements well DISIR and DISCA to avoid the agent having to browse the entire image. Moreover, the larger the image to be segmented, the more sense this approach makes. However, there is still room for improvement. Regarding the patch-based strategy, it could be useful to propose patches of variable sizes and of non-fixed locations, if the benefits outweigh the additional computational cost. Regarding the pixel-based strategy, confidence derivative maps could be also considered to instead target - or avoid - areas of confidence breakdown. Finally, further research distinguishing between epistemic (i.e. model uncertainty) and aleatoric (i.e. measurement uncertainty) uncertainties within our framework could provide a better understanding of the impact of annotations.



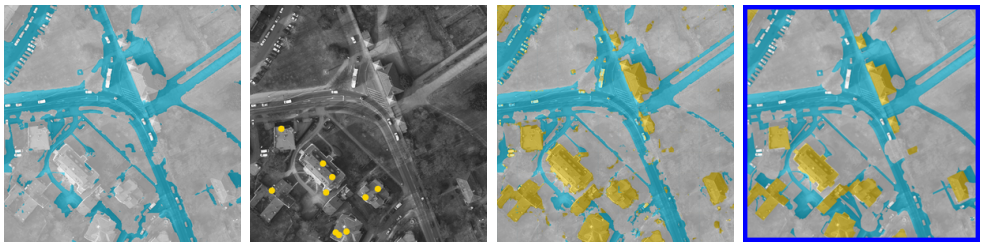
(a) Initial output (b) Entropy (c) Annotation(s) (d) DISCA output (e) Ground-truth

Figure 5.7 – Initial output corrected with annotations relying on entropy. On the "*Entropy*" column, the areas with an entropy higher than the ninth quantile over the image are highlighted in red. On the "*Annotation(s)*" column, the color of the annotations represents their labels w.r.t. the associated ground-truth maps.

Chapter VI - Towards interactive class-incremental segmentation

Contents

VI.A . Motivation & contributions



(a) Initial prediction (b) New class annotations (c) New prediction (d) Ground-truth

Figure 6.1 – Class-incremental use-case example : adding a *building* class to a *road* detector. In practice, it consists in inserting *building* in the {*background*, *road*} label space with clicked annotations on an image from the ISPRS dataset. Initial prediction (a) leads to new-class annotations to collect the new prediction (c).

Taking a step back, we have so far built an end-to-end framework for correcting segmentation maps interactively. It is composed of DISIR and DISCA, two different methodologies to process user inputs, and an active learning component to guide the user in their annotation task. Although these components can still be improved, they are ready for operational use. However, there remains an important limitation of our framework which has not been tackled yet : it is currently impossible to add classes. Indeed, this parameter is set before the initial training and cannot be changed afterwards. In this last chapter, our goal is to make this parameter modifiable in order to learn new segmentation classes from clicked user annotations.

Therefore, we are going to work on our third research axis : *How to adapt to new use cases ?*. We have already seen in Chapter IV that DISCA can effectively address domain adaptation issues. We now explore how to interactively add a new category to the classifier. Precisely, we focus on the modification of the neural network output space for interactive class-incremental learning.

A common issue of these algorithms is their lack of flexibility due to their frozen output space fixed before training [Tan et al., 2018, Panareda Busto and Gall, 2017]. A standard approach is then to keep the neural network, modify its architecture according to the new task (e.g. modify the head to add a possible label class) and fine tune it on the new task starting from its previous weights. However, it is not trivial since all weights are inter-connected in a neural network, which implies that the entire network needs to be retrained. Hence, due to both catastrophic forgetting and lack of convergence threats, fine-tuning a deep network on a small set of samples of new classes seems quite risky. However, we show here that this is possible within the right framework, as showed by Figure 6.1. .

To improve the neural networks plasticity for semantic segmentation, we notably adapt different regularizations in order to fully benefit from the sparse annotations. We also explore a pseudo-label strategy in order to alleviate catastrophic forgetting and avoid to query annotations from the previous classes.

Part of the works presented in this chapter have been published in **Weakly-supervised continual learning for class-incremental segmentation** Lenczner G., Chan-Hon-Tong A., Luminari N., & Le Saux B., IGARSS 2022 and the associated code is available on this GitHub repository : https://github.com/alteia-ai/ICSS.

VI.B . Methodology

VI.B.1 . Formalization, baseline and constraints of the problem

The typical problem we address here is the following. A neural network is initially trained for semantic segmentation to detect $N-1$ classes. It is then applied on a new image I. The goal is to add a new possible segmentation class and to provide an accurate segmentation map with respect to these N classes. To this aim, M annotations are provided to make the network interactively learn the N^{th} segmentation class.

In practice, a simple baseline to address this problem is *fine-tuning*, that is modifying the last layer of the neural network to increment its output space, and retraining it with a cross-entropy loss on the annotated pixels. However, different problems then arise :

— The **forgetting of previous classes** since the provided labels should only belong to the new class for a smooth process. Indeed, it would be counter-productive to query labels that are supposed to be learned already.

— A **background shift** [Cermelli et al., 2020] since the new class is usually previously classified as background. So, the network has to both learn to detect a new class and to modify the learned representation of the background class. It is also possible that the new class emerges from another class than background. For example, a "vegetation" class could be divided into a "tree" class and a "low vegetation" class. However, the problem is still similar, as a learned class has to be split into two and the original representation of that class has to be modified.

— The network has to learn a new class representation from an highly **sparse ground-truth** coming from the few clicked annotations.

— As before, due to the interactive perspective, time and complexity are still important.

The first two issues come directly from the class incremental problem while the third and the fourth ones comes from the interactive context in which we place ourselves. We are now going to detail how we address them. We notably rely on a pseudo-label strategy to prevent the forgetting of previous classes and adapt different regularizations added to the cross-entropy loss to better learn from the sparse annotations. We will see that the choice of regularization also depends on the additional computation time.

Following [Cermelli et al., 2020, Michieli and Zanuttigh, 2021] notation, we refer through the rest of this chapter to the previous network with the $N-1$ class prediction as the *old network* and to the new one as the *new network*. We also refer to the set of semantic classes except the background as the *classes of interest*.

VI.B.2 . Learning from the old network with pseudo-labeling

To prevent the forgetting of previous classes, we propose to rely on a pseudo-labeling strategy [Michieli and Zanuttigh, 2021]. Indeed, since they should have been learned by the old network, it is possible to use its predictions as noisy labels to retrain the new network for these classes. This allows to not require annotations belonging to these classes, which would be very inconvenient. To automatically sample the pseudo labels, we select the ones minimizing the entropy at the output of the softmax. This acts as a confidence proxy, similarly to what we have seen in Chapter V. We choose to sample a number of such annotations per class equivalent to the number of provided new-class annotations to keep a balanced training set.

However, since previous background labels would possibly belong to the new

class, background annotations can't be simulated this way. Indeed, since the background is split into two classes (a new background and the new class), it can also be seen as a new class. Therefore, the agent has to provide annotations for both the background and the new class while the annotations for the other classes of interest are sampled from the prediction of the old network.

VI.B.3 . Regularizations

Pseudo labeling is already a way to control the catastrophic forgetting. Yet, this can be combined with the addition of a regularization term. Hence, to mitigate catastrophic forgetting and make the most of the sparse provided annotations, we consider the following regularizations to add to the classic cross entropy loss. Two of them are distillation losses, as they are designed to transfer information between the old network and the new network. Another aims to optimally organize the latent space of the neural network and the last one enforces local spatial a prioris. We note $\mathbf{f}$ and $\mathbf{g}$ to represent respectively the old and the new networks, parameterized respectively by $\theta_{\mathbf{f}}$ $\theta_{\mathbf{g}}$ and $\mathbf{x}$ their inputs.

VI.B.3.a ODL

Since the new class was previously part of the background, the predictions over the other classes of interest should remain similar. To enforce this property, we draw inspiration from the regularization used with DISCA learning scheme in Chapter IV and consider adding a cross entropy regularization term over the pixels which are predicted as belonging to classes of interest by the old network. This overlaps with the pseudo-label strategy and can be seen as an output-level knowledge distillation loss, we thus call it Output Distillation Loss (ODL).

Formally, with the initial prediction $\mathbf{p_0} = f(\mathbf{x}, \theta_{f_0})$, c_0 the background class and where $\mathbf{1}$ represents the indicator function, we write :

$$\mathcal{L}_{\text{ODL}}(\mathbf{x}; \theta_{\mathbf{g}}, \theta_{\mathbf{f}}) = \| \left(\mathbf{g}(\mathbf{x}; \theta_{\mathbf{g}}) - \mathbf{f}(\mathbf{x}, \theta_{\mathbf{f}})\right) \mathbf{1}_{[\text{argmax } (\mathbf{f}(\mathbf{x}, \theta_{\mathbf{f}})) \neq c_0]} \|_1 \tag{6.1}$$

VI.B.3.b PodNet

Following PodNet [Douillard et al., 2020], we aggregate features over each of the spatial dimensions (H, W) to enforce statistic matches at the encoder level between the old network and the new one with a L^2 intermediate-level knowledge distillation loss. Formally, with C the number of channels and $\widetilde{\mathbf{f}}, \widetilde{\mathbf{g}}$ the encoders of

respectively $\mathbf{f}$ and $\mathbf{g}$, we write :

$$\begin{aligned}\mathcal{L}_{\text{PodNet}}(\mathbf{x};\theta_{\mathbf{g}},\theta_{\mathbf{f}}) =& \frac{1}{C\cdot W}\sqrt{\sum_{c=1}^{C}\sum_{w=1}^{W}\left(\sum_{h=1}^{H}\widetilde{\mathbf{g}}(\mathbf{x};\theta_{\mathbf{g}})_h-\sum_{h=1}^{H}\widetilde{\mathbf{f}}(\mathbf{x};\theta_{\mathbf{f}})_h\right)^2_{w,c}} \\ &+\frac{1}{C\cdot H}\sqrt{\sum_{c=1}^{C}\sum_{h=1}^{H}\left(\sum_{w=1}^{W}\widetilde{\mathbf{g}}(\mathbf{x};\theta_{\mathbf{g}})_w-\sum_{w=1}^{W}\widetilde{\mathbf{f}}(\mathbf{x};\theta_{\mathbf{f}})_w\right)^2_{h,c}}\end{aligned} \tag{6.2}$$

VI.B.3.c Consistency

Inspired by the success of consistency regularizations in semi supervised learning [Ouali et al., 2020, French et al., 2020], we force the network to make similar predictions on the original image and on a augmented version of it. In our case, we use simple data augmentation like horizontal flip rather than more custom ones like CutMix [French et al., 2020], as it achieved better results in this peculiar context. With $\mathbf{m}$ a geometrical augmentation function (e.g. horizontal flip) and $\mathbf{m}^{-1}$ its reciprocal function, we write :

$$\mathcal{L}_{\text{cons}}(\mathbf{x};\theta_{\mathbf{g}}) = \|\mathbf{g}(\mathbf{x};\theta_{\mathbf{g}}) - \mathbf{m}^{-1}\left(\mathbf{g}(\mathbf{m}\left(\mathbf{x}\right);\theta_{\mathbf{g}})\right)\|_2^2 \tag{6.3}$$

If the augmentation is not geometric (e.g. color jittering) and has no reciprocal function, we simply write :

$$\mathcal{L}_{\text{cons}}(\mathbf{x};\theta_{\mathbf{g}}) = \|\mathbf{g}(\mathbf{x};\theta_{\mathbf{g}}) - \mathbf{g}(\mathbf{m}\left(\mathbf{x}\right);\theta_{\mathbf{g}})\|_2^2 \tag{6.4}$$

VI.B.3.d SDR

Rooted in few-shots and contrastive learning, we draw inspiration from Sparse and Disentangled Representations (SDR) [Michieli and Zanuttigh, 2021] to organize the neural network latent space. This aims to reduce forgetting whilst improving the recognition of the new class. Concretely, it adds a prototype-based regularization at the encoder level of the neural network. Prototypes are vectors representative of each segmentation class and are computed on the fly at each new learning step. Formally, at training step t and for a segmentation class c, a prototype $\mathbf{p}$ is initialized as $\mathbf{p}_c(0) = \mathbf{0}$ and is defined as :

$$\mathbf{p}_c(t) = \mathbf{p}_c(t-1) + \frac{1}{\|\mathbf{1}_{[\mathbf{g}(\mathbf{x};\theta_{\mathbf{g}})=c]}\|_1}\left(\sum_{w=1}^{W}\sum_{h=1}^{H}\widetilde{\mathbf{g}}(\mathbf{x};\theta_{\mathbf{g}})_{h,w}\mathbf{1}_{[\mathbf{g}(\mathbf{x};\theta_{\mathbf{g}})=c]}\right)$$

With $\widetilde{\mathbf{y}}$ the down-sampled annotated label map matching the spatial dimensions of $\mathbf{g}(\mathbf{x};\theta_{\mathbf{g}})$, this regularization is then composed of three terms :

— **Matching term** : The updated prototypes must be close to the previous ones.

$$\mathcal{L}_{\text{match}}(t) = \frac{1}{N}\sum_{c=1}^{N}\frac{1}{\text{Card}(\mathbf{p}_c(t))}\|\mathbf{p}_c(t) - \mathbf{p}_c(t-1)\|_2$$

— **Repulsive term** : The different prototypes must be far from each other.

$$\mathcal{L}_{\text{repuls}}(t) = \frac{1}{N}\sum_{c=1}^{N}\frac{1}{\|\mathbf{1}_{[\widetilde{\mathbf{y}}=c]}\|_1}\sum_{\widetilde{c}=1,\widetilde{c}\neq c}^{N}(\|\mathbf{p}_c(t) - \mathbf{p}_{\widetilde{c}}(t)\|_2)^{-1}$$

— **Attraction term** : The pixels must be close to their associated prototype.

$$\mathcal{L}_{\text{attract}}(t) = \frac{1}{N}\sum_{c=1}^{N}\frac{1}{\|\mathbf{1}_{[\widetilde{\mathbf{y}}=c]}\|_1}\|\left(\widetilde{\mathbf{g}}(\mathbf{x};\theta_{\mathbf{g}}) - \mathbf{p}_c(t)\right)\cdot\mathbf{1}_{[\widetilde{\mathbf{y}}=c]}\|_2$$

Finally :

$$\mathcal{L}_{\text{SDR}}(t) = \mathcal{L}_{\text{match}}(t) + \mathcal{L}_{\text{repuls}}(t) + \mathcal{L}_{\text{attract}}(t) \tag{6.5}$$

VI.B.3.e FESTA

To deal with sparse annotations in remote sensing semantic segmentation, the FEature and Spatial relaTional regulArization (FESTA) [Hua et al., 2021] is an unsupervised loss that accounts for neighborhood structures both in spatial and feature domains, as it assumes that nearby pixels share labels.

Precisely, it acts in the feature space just before the softmax layer on a random set of pixels P. First, for each of these pixels x_p, it strengthens in the feature space both a rapprochement between the pixel (noted $\widetilde{x}_p$ in the feature space) and its neighbor $\widetilde{x}_{p\star}$ and a distance between the pixel and the one that differs the most $\widetilde{x}_{\overline{p}}$. Second, since segmentation maps are usually spatially continuous, it assumes that there is necessarily a pixel among the neighbors to share the same label class. Hence, it also enforces a rapprochement between the pixel and one spatial neighbor among the eight neighbors. The chosen spatial neighbor $\widetilde{x}_{\hat{p}}$ is the closest one in the feature space in order to ensure that it belongs to the same class. Formally, with S representing the cosine similarity, we write :

$$\mathcal{L}_{\text{FESTA}} = \frac{1}{P}\left(\alpha\sum_{p=1}^{P}\|\widetilde{x}_p - \widetilde{x}_{p\star}\|_2 + \beta\sum_{p=1}^{P}\mathcal{S}\left(\widetilde{x}_p, \widetilde{x}_{\overline{p}}\right) + \gamma\sum_{p=1}^{P}\|\widetilde{x}_p - \widetilde{x}_{\hat{p}}\|_2\right) \tag{6.6}$$

Following the original paper [Hua et al., 2021], α, β and γ are respectively set to 0.5, 1 and 1.5 in our experiments.

VI.C . Experiments

Except when specified otherwise, we simulate $N = 300$ new-class (*building*) and background annotations. Regarding the old class (*road*) pseudo-labels, we study two strategies :

— *Sparse pseudo-labels :* We simulate an equivalent number N of *road* sparse pseudo-labels by selecting the most confident predicted pixels.

— *Full pseudo-labels :* We select all the predicted labels with a *road* probability superior to $\delta = 0.95$ after the softmax layer and weight the cross-entropy loss with frequency balancing.

For all experiments, we use a LinkNet [Chaurasia and Culurciello, 2017] architecture trained using Adam optimizer with a learning rate of 10^{-4} during 10 pseudo-epochs. Each pseudo-epoch consists in $10\,000$ 256×256 labeled samples randomly chosen from training data. We infer using a 256×256 sliding window with an overlap of 50%. Due to the stochastic nature of the optimization process and the simulation of the annotations, all experiments are averaged on 3 runs to obtain more statistically significant results.

During fine-tuning, we use an Adam optimizer with a learning rate of $2 \cdot 10^{-5}$. Each training step consists of 10 back-propagation iterations. We fine-tune for 30 training steps and select the best performances obtained in the last 15 steps. In a real use case (i.e. without access to the whole ground truth and therefore without the possibility to determine the best performances), this would correspond to a user manually choosing a result among a list of proposals. For comparison, we will also analyze the results obtained at the different training step.

To compare to a potential upper bound, we consider control networks directly pretrained on dense {*road, building*} ground-truth maps (and not fine-tuned).

VI.C.1 . Approach assessment

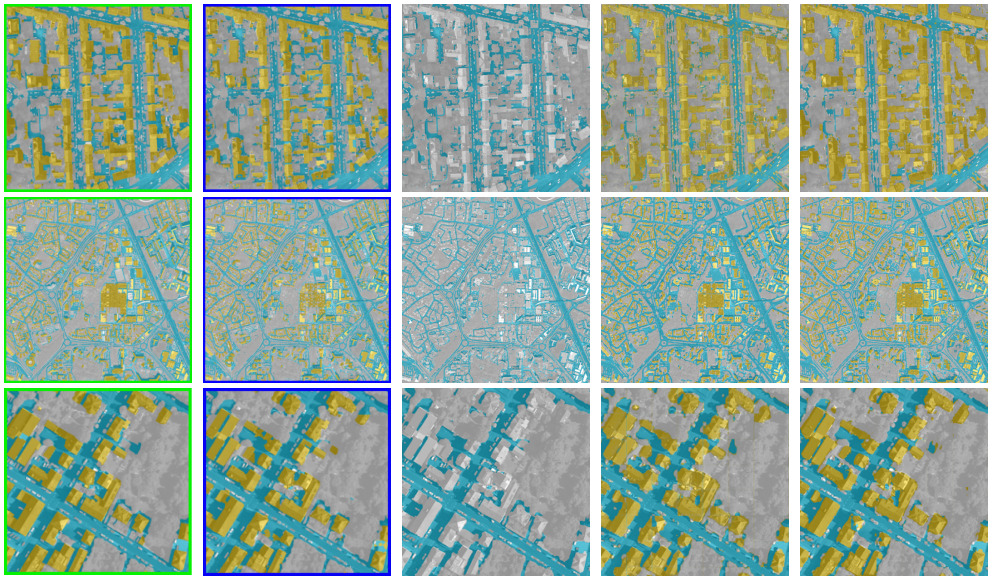
	Control	Baseline	FESTA	ODL	PodNet	SDR	Cons.
$\text{ISPRS}_{Pot.}$	76.4	$68.7^{+- 2.7}$	$66.5^{+- 8.4}$	$68.2^{+- 4.1}$	$71.7^{+- 2.3}$	$72.0^{+- 0.3}$	**$74.8^{+- 3.1}$**
$\text{ISPRS}_{Vai.}$	84	$76.2^{+- 2.3}$	$75.1^{+- 3.4}$	$74.8^{+- 1.8}$	$74.1^{+- 5.4}$	**$79.7^{+- 1.6}$**	$77.6^{+- 4.7}$
SemCity	74	$63.2^{+- 3.2}$	$62.4^{+- 1.5}$	$54.0^{+- 0.9}$	$65.4^{+- 1.0}$	**$67.7^{+- 0.6}$**	$50.5^{+- 2.1}$

Table 6.1 – Comparison (IoU) of the different regularizations with sparse pseudo labels.

As indicate Table 6.1 and Table 6.3, the neural network is effectively able to learn a new class with clicked annotations as ground-truth.

VI.C.1.a Sparse pseudo-labels strategy

Even without additional regularizations, it reaches an IoU over the three classes of 68.7% on Potsdam, 76.2% on Vaihingen and 63.2% on Toulouse. Appropriate regularization further improves the performances up to 4%. Indeed, SDR consistently



(a) Ground-truth (b) Control prediction (c) Old prediction (d) Baseline prediction (e) SDR prediction

Figure 6.2 – Building and road segmentation with 300 building annotations and road pseudo-labels on an example from Toulouse and Vaihingen compared to ground-truth and control prediction.

improves the results on the three datasets : 3.3% on Potsdam, 3.5% on Vaihingen and 4.5% on Toulouse. Moreover, SDR tends to stabilize the results compared to the baseline, as testifies the standard deviation measures (e.g. the std on Toulouse is of 0.6 for SDR and of 3.2 for the baseline). As shows Figure 6.2, it seems to visually translate into sharper contours, even though baseline already produces visually accurate results. The consistency regularization reaches even higher performances on Potsdam (+6.1% w.r.t. the baseline) but is less stable than SDR and it is not consistent on the other datasets. The other regularizations under-perform in this setting in regards to SDR and can even lead to worse results than the baseline on Vaihingen and Toulouse. Hence, the distillation losses and FESTA seem to be less relevant for this problem than the latent space regularization proposed by SDR. We have also tried to combine by pairs the different regularizations but this did not lead to an increase on performances compared to SDR alone.

As shown by Table 6.2, the good performances with SDR are almost equivalent between the three classes. Interestingly, the "*road*" class is the one with the worse results (65.8% IoU in average over the three datasets), compared to "*background*" (76.3% IoU) and "*buildings*" (72.2% IoU). This is also lower than with the old network, which respectively reaches 76.73%, 78.6% and 65.2% on Potsdam, Vaihingen and Toulouse on the road class (i.e. 73.5% in average). This drop of performances is particularly strong on Potsdam (-12%) and it is probably due to the use of point pseudo labels obtained with the old network that poorly capture the semantics of

the objects. To address this issue, we now consider the full pseudo label strategy.

	Background	Roads	*New* Buildings	Δ_{roads}(SDR-old)
ISPRS$_{Pot.}$	$74^{+-0.4}$	$64.7^{+-0.8}$	$77.2^{+-0.3}$	-12
ISPRS$_{Vai.}$	$84.8^{+-1.5}$	$74.6^{+-0.9}$	$79.9^{+-2.3}$	-4
SemCity	$72.9^{+-0.7}$	$63.9^{+-0.8}$	$66.2^{+-1.1}$	-1.3

Table 6.2 – IoU per class with SDR regularization on sparse pseudo label. For comparison, we also provide the IoU difference between the new network with SDR and the old network on the *road* class.

VI.C.1.b Full pseudo labels

	Control	Baseline	FESTA	ODL	PodNet	SDR	Cons.
ISPRS$_{Pot.}$	76.4	$74.5^{+-2.8}$	$48.7^{+-12.3}$	$70.5^{+-2.7}$	$75.3^{+-4.7}$	$75.9^{+-1.7}$	$\mathbf{76.9^{+-2.2}}$
ISPRS$_{Vai.}$	84	$77.1^{+-1.2}$	$68.4^{+-3.4}$	$76.4^{+-1.6}$	$75.1^{+-4.9}$	$\mathbf{79.9^{+-1.5}}$	$77^{+-4.1}$
SemCity	74	$62.7^{+-1.7}$	$49.2^{+-12.8}$	$59.6^{+-0.8}$	$61.6^{+-1.9}$	$\mathbf{65.2^{+-0.6}}$	$60.7^{+-5.1}$

Table 6.3 – Comparison (IoU) of the different regularizations with full pseudo labels.

As we can observe on Table 6.3, the full pseudo label strategy improves the results on the two ISPRS datasets : up to 4% on Potsdam and 3% on Vaihingen. Except for FESTA which leads now to deceiving results on all datasets, the results of the different regularizations remain consistent with the ones from the sparse pseudo label strategy. Indeed, SDR is still the most stable and performing regularization, except on Potsdam where the consistency regularization still leads to better (but a bit more unstable) results. As we can observe on Table 6.4, the drop of performances that was observed on the Potsdam dataset with sparse pseudo labels is mostly fixed with the full pseudo labels (from -12% to -2.7%).

Interestingly, the results obtained on SemCity Toulouse are worse than with sparse pseudo-labels (e.g. -2.5% with SDR). This is probably because the old network is less efficient than on the other datasets : 88.9% of the pseudo-labels on SemCity Toulouse are well classified while 97.5% are well classified on Potsdam. Hence, using full pseudo labels implies using false ones and thus adding too much noise during fine-tuning.

Therefore, when the old network produces mitigated pseudo-labels, it may be better to use sparse pseudo-labels. In this case, it may also be too ambitious to try to increase the label space and it would be better to first correct the labels of existing classes, potentially with DISIR and DISCA. However, when the old network produces high quality labels, the use of full pseudo labels with a weighted cross

entropy allows to keep good performances on the old classes and efficiently learn the new classes.

	Background	Roads	*New* Buildings	Δ_{roads}(SDR-old)
$ISPRS_{Pot.}$	$76.1^{+-1.3}$	$74^{+-1.2}$	77.7^{+-3}	-2.7
$ISPRS_{Vai.}$	$84.9^{+-1.2}$	$73.6^{+-2.8}$	$81.3^{+1.2}$	-5.2
SemCity	$69.1^{+-1.5}$	$61.4^{+-0.7}$	$65.2^{+-1.5}$	-3.8

Table 6.4 – IoU per class with SDR regularization on full pseudo-labels. For comparison, we also provide the IoU difference between the new network with SDR and the old network on the *road* class.

VI.C.1.c Results at the different training steps

For comparison, Figure 6.3 presents the results obtained at the 15 last training steps instead of the best one. FESTA and ODL are the most unstable and they undergo an average decrease of respectively 6.2% and 10.6% IoU between the last iteration and the best one. SDR brings consistency again and still provides the best results. Its average decrease is only of 1.7%, even though it is a bit more unstable on the ISPRS Potsdam dataset. The reason is potentially that this regularization acts on the latent space of the neural network, since PodNet also acts there and is also stable and prevents a significant drop (1.97% in average). The baseline is a bit more unstable than PodNet and SDR and it undergoes an average decrease of 3.5% between the last iteration and the best one.

Therefore, this shows that making an agent choose the best results is relevant but also that the networks are robust and consistent despite the sparse training data when choosing the right regularization. Hence, arbitrarily choosing the results does not lead to a catastrophic drop of performances, especially with regularizations acting on the latent space of the neural network.

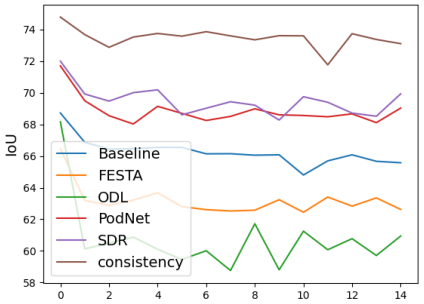
VI.C.2 . Freezing the network

	No freeze	Encoder only	All	From scratch
$ISPRS_{Potsdam}$	**72**	68.5	47.78	39.25

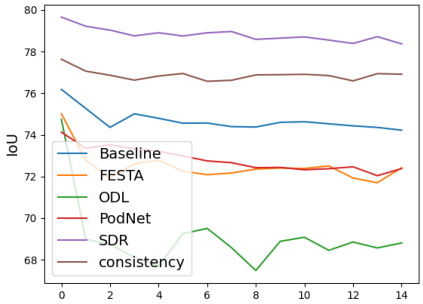
Table 6.5 – IoU when freezing different parts of the network and when training from scratch.

We also investigate the impact of freezing of different parts of the network during the interactive retraining. Specifically, we consider freezing :

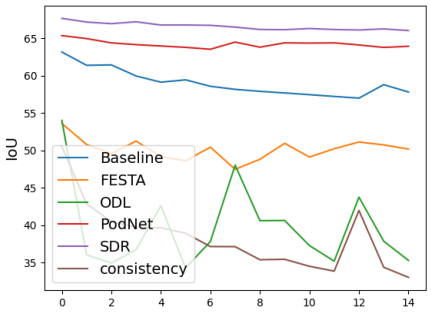
— All the layers but the last one



(a) ISPRS$_{Pot.}$



(b) ISPRS$_{Vai.}$



(c) SemCity

Figure 6.3 – IoU evolution over the last 15 training iteration

— Only the encoder layers of the encoder/decoder architecture

In these settings, regularizations like SDR or PodNet, which act on the latent representation space of the encoder, are not relevant anymore. We compare these settings to freezing no layers at all, as we did in the other experiments. Finally, we also consider training a neural network from scratch on the sparse clicked annotations instead of fine-tuning a copy of the old network to evaluate its impact.

As shows Table 6.5, freezing even partially the network leads to sub-optimal results : if freezing the encoder already leads to a 3% drop, the freezing of the entire network leads to a drop of more than 20% IoU. Therefore, this freezing constraint is probably too strong and therefore limits the neural network plasticity needed to learn a new class.

The training from scratch also leads to significantly worse results : the performances are almost cut in half compared to a fine-tuning of the old network. Therefore, even without taking into account its utility for the pseudo-labels, it appears absolutely necessary to fine-tune a copy of the old network instead of training a neural network from scratch.

VI.C.3 . Influence of the number of annotations

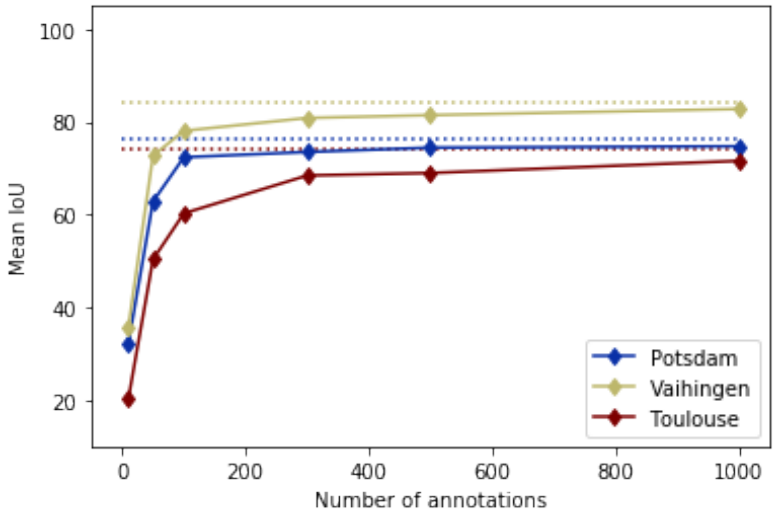


Figure 6.4 – IoU evolution on the three datasets with respect to the number of annotations. Control network performances are in dots.

To better apprehend our approach, we finally analyze the influence of the number of annotations over the performances. To this purpose and according to our previous results, we fine-tune the network using SDR regularization.

As we can observe on Figure 6.4, unlike when fine-tuning on already learned classes in Chapter IV, a low number of annotations (e.g. less than one hundred) leads here to sub-optimal results. However, the performances then greatly improve with respect to the number of annotations : they increase from 29% IoU in average on the three datasets for 10 annotations to 67 % IoU for 100 annotations. Notably, after 100 annotations for the two ISPRS datasets and 300 annotations for the SemCity dataset, the neural networks almost catch up with the control network performances but they then stabilize on a plateau. These plateau values probably show the limits that can be reached under such a training regime.

VI.C.3.a Computational times

Baseline	FESTA	ODL	PodNet	SDR	Cons.
190	520	190	330	500	330

Table 6.6 – Computational time (s) for one learning step on a 6000×6000 image depending on the regularization. Each of our learning steps represent 10 back-propagation iterations.

Since our goal is interactivity, we also compare the computation times associated with the different regularizations in Table 6.6. The good performances and the

stability of SDR are then weighted by its computing time : it takes 2.5 more times (520s) than without any regularization (190s). However, thanks to its stability, training with SDR may therefore require less learning steps, which would balance the longer computation time. The benefit versus the added cost will therefore depend on the use cases and users.

VI.D . Conclusion

Relying on relevant regularizations and on pseudo-labeling to mitigate catastrophic forgetting and avoid to query annotations from the previous classes, we have shown that it is possible to make a neural network learn on the fly new segmentation classes with point supervision after the initial training. Notably, the SDR regularization, relying on a latent space layout, seems particularly suitable for this task, even though it considerably increases the processing time. Indeed, not only does it consistently improve results, it also stabilizes the convergence of the neural networks in this context. Moreover, to optimally benefit from this approach, no layers of the network should be frozen and it is mandatory to fine-tune a copy of the old network and not to train a neural network from scratch. Finally, regarding the amount of annotations, it is necessary to gather approximately one hundred annotations per tile for the success of the proposed methodology.

Indeed, unlike the refinement of already known segmentation classes as in Chapter IV which did not require many annotations, a low number of annotations does not allow to learn the new segmentation class. On the other hand, a thousand clicked annotations does not significantly improve the results : this shows the limits of the results that can be obtained with this methodology. This required number of annotations for optimal performances is limiting in an interactive context. To improve this approach, it could be possible to draw inspiration from FixMatch [Sohn et al., 2020] or CutMix-Seg [French et al., 2020] with a second stage of pseudo-labeling to also sample new class annotations.

It would also be interesting to test other use cases, especially by working with other segmentation classes, like vegetation classes. It would also be very interesting for the community to add domain adaptation to be in an open-set learning paradigm. However, this would not be trivial at all, especially since the pseudo-labels obtained with the old network would be noisier.

Chapter VII - Conclusion

Contents

When this PhD began in 2019, deep neural networks had been the state-of-the-art algorithms in computer vision and its application areas for several years already and both theoretical and practical deep learning foundations were already well established.

That being said, current research often aims to bridging the gap between excellent performances and human-level decisions. It includes robustness, fairness and explainability, but also prominently *learning better with less annotated data* with various fields such as few shot learning, active learning, weakly supervised learning or semi-supervised learning. Following this line of paradigms, our overall goal was to build a synergy between neural networks for semantic segmentation and their users to smoothly collect semantic segmentation label maps. In this perspective, we have built throughout this thesis a general framework to incrementally refine segmentation map proposed by a neural network. This framework relies on different components, each described in a specific Chapter in this thesis. These components also all address one of the research questions detailed in Introduction. We now review these research questions to see the answers provided during this thesis.

VII.A . Summary of contributions

VII.A.1 . How to interact with neural networks after learning ?

The typical use of DNNs is to train them offline on fully annotated data and then to deploy them automatically on new data. However, this is often unsuitable in industry due to potential errors. Assuming a human operator necessary to assert the results, our goal was then to interactively correct segmentation maps output by DNNs after training with the following constraints.

— The user inputs have to be easy and intuitive : the user should not be put off by the framework.
— The interactions must be smooth and fast so it is not possible to keep training data in memory or to re-train the algorithm entirely.
— The user inputs should generalize as well as possible to the whole scene under study.

To address these issues, we have proposed two complementary **interactive learning methodologies** relying on clicked user inputs to refine semantic segmentation maps. In both methodologies, the user clicks on mislabeled areas on the segmentation map and these clicks are then used by the neural network to modify its initial prediction. DISIR, presented in Chapter III, only modifies the inputs of the neural network for fast but local corrections. The clicks are first encoded into N annotation channels with distance transform where N is the number of segmentation classes. These channels are then concatenated with the RGB image at the input of the network. DISCA, presented in Chapter IV, modifies the weights of the neural network with on-the-fly retraining for slower but more global corrections. Key to avoid over-fitting on the annotations is a regularization mechanism based on the initial prediction of the DNN. We have also investigated Transformer architectures and their self-attention mechanism with DISIR in order to go beyond the spatial limitation induced by convolutional receptive fields without requiring a retraining step. However, this last idea has yielded ambiguous results.

VII.A.2 . How to get relevant data ?

Since remote sensing images can be huge, it can be tedious for an operator to review the associated segmentation maps in detail. It can then be useful to guide the user to relevant areas to annotate but the question of what is relevant data then arises. To optimize the training sets in active learning, data selection usually relies on the uncertainty of the model, on the representativeness of the data with respect to the dataset or on a combination of both. Since we aim to spot wrong predictions and not to diversify our training set, we have chosen to focus on uncertainty measures instead of representativeness ones for data selection to guide the user.

Hence, our general goal still being to ease the burden of the operator, we have proposed an **active learning** scheme to guide the annotator towards relevant data to examine. DIAL, presented in Chapter V, relies on different uncertainty measures to guide the user towards areas to correct using either DISIR or DISCA. It uses uncertainty to present the user with both relevant patches to annotate and uncertainty maps for a guidance at the pixel level. Notably, entropy and ConfidNet

measures are the most relevant ones as they are both efficient and fast to compute.

VII.A.3 . How to adapt to new data and new use cases ?

A known problem of DNNs is that they are prone to overfit on training data. This means that it can be difficult to apply them to new unseen data with different data distribution (i.e. domain adaptation) or with a different label space.

We have first shown in Chapter IV that DISCA can be effectively applied to domain adaptation issues. Indeed, it enables a quick adaptation of a DNN to new data following unseen distribution. Hence, only retrained on a few user annotations, a DNN is able to output segmentation maps almost equivalent to the ones output by a DNN classically trained on fully annotated data from the same distribution as the new data.

We have then further explored **continual learning** in Chapter VI by learning a new segmentation class. This necessarily implies to modify the DNN architecture and this also raises two additional main issues. First, the semantic of the new class has to be learned under weak supervision since there are only sparse clicked inputs. Second, the previously learned classes must not be forgotten by the algorithm.

To optimize the learning on the sparse annotations, we have studied several regularizations added to the cross entropy loss. The ones based on the organization of the latent space of the DNN and on the consistence of the predictions across different data augmentations show the best results. To prevent the forgetting of the previous classes, we have relied on a pseudo label strategy to sample labels from the old classes from the most confident predictions of the old network.

Given a sufficient amount of user annotations, the proposed incremental methodology has shown promising results on urban landcover applications.

VII.A.4 . User adoption

These algorithms and methodologies answer to internal use-cases at Alteia and ONERA. We present some of them :

- DISIR and DISCA are used to segment stockpiles in quarries to then estimate their volumes. There may then be problems of domain adaptation between the different quarries.
- The class incremental methodology can be used to differentiate the species detected by a tree detector for forest or crop monitoring.

Moreover, the open-source release of our codes for research has led to the adoption of our algorithms by other users [Dumas et al., 2022]. Notably, a commercial license to a company in the aerospace sector is currently being negotiated.

VII.B . Future works

We have therefore provided an answer to each of our research questions, proposed a fairly complete framework and experimentally validated each of its components. Still, various ideas could continue the presented work and lead to

further improvements.

VII.B.1 . How to interact with neural networks after learning ?

VII.B.1.a Short-term improvements

Regarding the user inputs, we have made the choice during this thesis to only consider point clicked annotations. Although this form of annotation is predominant in many interactive works and relevant because of its ease of use, other more flexible forms could be explored in our framework. It could be first extended to lines or scribbles annotations. This would require to modify the encoding of the annotations and to carefully redesign the sampling of the annotations from the ground-truth during training.

An interesting alternative would be to consider polygon outputs instead of segmentation masks. As introduced by Polygon RNN [Acuna et al., 2018] and Curve GCN [Ling et al., 2019], the user interactions would then modify these polygons.

VII.B.1.b Long-term improvements

A more ambitious prospect would be to completely change the nature of the user inputs. In line with pioneering works on Visual Question Answering (VQA) for remote sensing [Chappuis et al., 2020], we can then imagine a more user-friendly interface with textual interactions. While that would require to also leverage powerful text models, recent architectures like Perceiver [Jaegle et al., 2022] or DALL-E [Ramesh et al., 2021] are already paving the way towards extremely strong multi-modal models. The clicked user inputs could be also combined with crowd-sourced data. For instance, [Sunkara et al., 2020] collect geo-localized tweets to help to evaluate the destruction caused by floods in climate-vulnerable regions. Clicked user inputs could bring additional information when crowd-sourced data and labels are missing while textual inputs could better propagate the sparse and localized clicks.

VII.B.2 . How to get relevant data ?

VII.B.2.a Short-term improvements

In Chapter V, we have proposed to guide the users towards relevant patches to annotate based on uncertainty measures.

First, it could be interesting to explore other kinds of measurements by focusing less on the model and more on the data, for instance with representativeness measures [Sener and Savarese, 2018]. Some simpler heuristics such as the entropy of the data or the cluster-based procedure proposed by [Dasgupta and Hsu, 2008] that leverages the natural hierarchical structure of remote sensing data could also be relevant to guide the user only according to the data.

Second, the patches are currently not chosen freely but from a grid. Choosing the patches freely in the image could be valuable since the grid may split the image in a sub-optimal way. However, this would not be trivial as it would lead to additional calculations with patch size parameters to be tuned and overlapping problems between the patches to be managed.

VII.B.2.b Long-term improvements

Reinforcement learning is a training paradigm in machine learning based on rewarding desired behaviors and punishing undesired ones. Like [Liu et al., 2019, Fang et al., 2017], building an acquisition function based on reinforcement learning to select the queries could allow the model to decide for itself the relevance of the action with respect to the data to better adapt to unknown and changing environments. This improved feedback from the algorithms could then lead them to work in better synergy with their user.

VII.B.3 . How to adapt to new data and new use cases ?

VII.B.3.a Short-term improvements

To get a broader view of the potential of our work as a whole, it could be used to tackle new use cases.

First, from a user-oriented perspective, our work could be a useful tool for Earth sciences applications where annotated data are missing, such as iceberg and sea-ice mapping, plastic debris detection or high resolution burned area detection. This would probably involve working beyond optical data, such as with multi-spectral images or digital surface models (DSMs).

Second, the idea of refining from a simple click to obtain a spatialized output is also present in video where many works [Vaudaux-Ruth et al., 2021, Alwassel et al., 2018, Derpanis et al., 2012] try to make temporal detection of action from an instant internal to the action (spot frame). Using the click of an operator would then be a direct extension of this work in interactive learning.

Finally, it would be useful to improve the class incremental approach proposed in Chapter VI when dealing with a low number of new class annotations. Drawing inspiration from semi-supervised approaches like FixMatch [Sohn et al., 2020] or CutMix-Seg [French et al., 2020] to improve the pseudo-labeling strategy to sample new class pseudo labels could be the key to solve this problem.

VII.B.3.b Long-term improvements

Drawing inspiration from large generative text models like GPT-3 [Brown et al., 2020], a possibility to address open-set learning in Earth Observation and quickly adapt models would be to pre-train in a self-supervised fashion a generic model on Sentinel data archives available through the satellite data archives (e.g. Data Integration and Analysis System (DIAS) archives) and then fine-tune it on downstream tasks.

Making progress on this topic would result in procedures to reconfigure generic-yet-efficient Artificial Intelligence for Earth Observation models rapidly to new use-cases. It would pave the way to artificial intelligence able to continually learn and accumulate knowledge, and so to adapt to the unforeseen.

Our aim during this research work was to lower the barrier between machine learning algorithms and Earth observation users. Advancing on these questions would further improve the collaboration between deep neural networks and their users, which is crucial to bridging the gap between purely academic works and real-world applications. Concretely, this would result in powerful deep learning models able to take multi-modal user inputs to quickly adapt to new use-cases. With such control over the models, experts, who are sometimes wary of fully automatic methods, could more easily adopt these user-centered methods to help them solve complex tasks.

Bibliographie

[Achanta et al., 2012] 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.

[Acuna et al., 2018] Acuna, D., Ling, H., Kar, A., and Fidler, S. (2018). Efficient interactive annotation of segmentation datasets with Polygon-RNN++. In *Conference on Computer Vision and Pattern Recognition*, pages 859–868. IEEE.

[Agustsson et al., 2019] Agustsson, E., Uijlings, J. R., and Ferrari, V. (2019). Interactive full image segmentation by considering all regions jointly. In *Conference on Computer Vision and Pattern Recognition*, pages 11622–11631. IEEE.

[Aksoy et al., 2004] Aksoy, S., Koperski, K., Tusk, C., and Marchisio, G. (2004). Interactive training of advanced classifiers for mining remote sensing image archives. In *Knowledge Discovery in Databases*, pages 773–782. ACM.

[Al Rahhal et al., 2022] Al Rahhal, M. M., Bazi, Y., Al-Dayil, R., Alwadei, B. M., Ammour, N., and Alajlan, N. (2022). Energy-based learning for open-set classification in remote sensing imagery. *International Journal of Remote Sensing*, pages 1–11.

[Alonso et al., 2021] Alonso, I., Sabater, A., Ferstl, D., Montesano, L., and Murillo, A. C. (2021). Semi-supervised semantic segmentation with pixel-level contrastive learning from a class-wise memory bank. In *International Conference on Computer Vision*. IEEE.

[Alwassel et al., 2018] Alwassel, H., Heilbron, F. C., and Ghanem, B. (2018). Action search : Spotting actions in videos and its application to temporal action localization. In *European Conference on Computer Vision*, pages 251–266. Springer.

[Andriluka et al., 2018] Andriluka, M., Uijlings, J. R., and Ferrari, V. (2018). Fluid annotation : A human-machine collaboration interface for full image annotation. In *MultiMedia (MM)*, pages 1957–1966. ACM.

[Audebert et al., 2016] Audebert, N., Saux, B. L., and Lefèvre, S. (2016). Semantic segmentation of Earth observation data using multimodal and multi-scale deep networks. In *Asian Conference on Computer Vision*, pages 180–196. Springer.

[Badrinarayanan et al., 2017] 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.

[Baetens et al., 2019] Baetens, L., Desjardins, C., and Hagolle, O. (2019). Validation of Copernicus sentinel-2 cloud masks obtained from MAJA, Sen2Cor, and

FMask processors using reference cloud masks generated with a supervised active learning procedure. *Remote Sensing*, 11(4) :433.

[Barbato et al., 2021] Barbato, F., Toldo, M., Michieli, U., and Zanuttigh, P. (2021). Latent space regularization for unsupervised domain adaptation in semantic segmentation. In *Conference on Computer Vision and Pattern Recognition*, pages 2835–2845. IEEE.

[Baumgardner et al., 2015] Baumgardner, M. F., Biehl, L. L., and Landgrebe, D. A. (2015). 220 band aviris hyperspectral image data set : June 12, 1992 indian pine test site 3. *Purdue University Research Repository*, 10 :R7RX991C.

[Bearman et al., 2016] Bearman, A., Russakovsky, O., Ferrari, V., and Fei-Fei, L. (2016). What's the point : Semantic segmentation with point supervision. In *European Conference on Computer Vision*, pages 549–565. Springer.

[Beluch et al., 2018] Beluch, W. H., Genewein, T., Nürnberger, A., and Köhler, J. M. (2018). The power of ensembles for active learning in image classification. In *Conference on Computer Vision and Pattern Recognition*, pages 9368–9377. IEEE.

[Benenson et al., 2019] Benenson, R., Popov, S., and Ferrari, V. (2019). Large-scale interactive object segmentation with human annotators. In *Conference on Computer Vision and Pattern Recognition*, pages 11700–11709. IEEE.

[Besnier et al., 2021] Besnier, V., Picard, D., and Briot, A. (2021). Learning uncertainty for safety-oriented semantic segmentation in autonomous driving. In *International Conference on Image Processing*, pages 3353–3357. IEEE.

[Boguszewski et al., 2021] Boguszewski, A., Batorski, D., Ziemba-Jankowska, N., Dziedzic, T., and Zambrzycka, A. (2021). Landcover. ai : Dataset for automatic mapping of buildings, woodlands, water and roads from aerial imagery. In *Conference on Computer Vision and Pattern Recognition*, pages 1102–1110. IEEE.

[Boykov and Jolly, 2001] Boykov, Y. Y. and Jolly, M.-P. (2001). Interactive graph cuts for optimal boundary & region segmentation of objects in nd images. In *International Conference on Computer Vision*, pages 105–112. IEEE.

[Brown et al., 2020] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33 :1877–1901.

[Bruzzone and Persello, 2009] Bruzzone, L. and Persello, C. (2009). Active learning for classification of remote sensing images. In *International Geoscience and Remote Sensing Symposium*, pages 689–693. IEEE.

[Bruzzone and Prieto, 1999] Bruzzone, L. and Prieto, D. F. (1999). An incremental-learning neural network for the classification of remote-sensing images. *Pattern Recognition Letters*, 20(11-13) :1241–1248.

[Carvalho et al., 2018] Carvalho, M., Le Saux, B., Trouvé-Peloux, P., Almansa, A., and Champagnat, F. (2018). On regression losses for deep depth estimation. In *International Conference on Image Processing*, pages 2915–2919. IEEE.

[Castillo-Navarro et al., 2019] Castillo-Navarro, J., Audebert, N., Boulch, A., Le Saux, B., and Lefèvre, S. (2019). What data are needed for semantic segmentation in Earth observation ? In *Joint Urban Remote Sensing Event*. IEEE.

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

[Castillo-Navarro et al., 2021b] Castillo-Navarro, J., Le Saux, B., Boulch, A., and Lefèvre, S. (2021b). Energy-based models in Earth observation : From generation to semi-supervised learning. *IEEE Transactions on Geoscience and Remote Sensing*.

[Caye Daudt et al., 2019] Caye Daudt, R., Le Saux, B., Boulch, A., and Gousseau, Y. (2019). Guided anisotropic diffusion and iterative learning for weakly supervised change detection. In *Conference on Computer Vision and Pattern Recognition Workshop*. IEEE.

[Cermelli et al., 2020] Cermelli, F., Mancini, M., Bulo, S. R., Ricci, E., and Caputo, B. (2020). Modeling the background for incremental learning in semantic segmentation. In *Computer Vision and Pattern Recognition*, pages 9233–9242. IEEE.

[Chappuis et al., 2020] Chappuis, C., Lobry, S., Kellenberger, B., Saux, B. L., and Tuia, D. (2020). How to find a good image-text embedding for remote sensing visual question answering ? In *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases MACLEAN Workshop*. Springer.

[Chaurasia and Culurciello, 2017] Chaurasia, A. and Culurciello, E. (2017). LinkNet : Exploiting encoder representations for efficient semantic segmentation. In *Conference on Visual Communications and Image Processing*. IEEE.

[Chen et al., 2021a] Chen, J., Lu, Y., Yu, Q., Luo, X., Adeli, E., Wang, Y., Lu, L., Yuille, A. L., and Zhou, Y. (2021a). TransUNet : Transformers make strong encoders for medical image segmentation. *arXiv preprint arXiv :2102.04306*.

[Chen et al., 2017] Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. (2017). Deeplab : Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4) :834–848.

[Chen et al., 2018] Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. (2018). Encoder-decoder with atrous separable convolution for semantic image

segmentation. In *European Conference on Computer Vision*, pages 801–818. Springer.

[Chen et al., 2019] Chen, Q., Wang, L., Wu, Y., Wu, G., Guo, Z., and Waslander, S. L. (2019). Aerial imagery for roof segmentation : A large-scale dataset towards automatic mapping of buildings. *ISPRS Journal of Photogrammetry and Remote Sensing*, 147 :42–55.

[Chen et al., 2020] 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.

[Chen et al., 2021b] Chen, X., Zhao, Z., Yu, F., Zhang, Y., and Duan, M. (2021b). Conditional diffusion for interactive segmentation. In *International Conference on Computer Vision*, pages 7345–7354. IEEE.

[Ciresan et al., 2012] Ciresan, D., Giusti, A., Gambardella, L., and Schmidhuber, J. (2012). Deep neural networks segment neuronal membranes in electron microscopy images. *Advances in Neural Information Processing Systems*, 25.

[Civco, 1993] Civco, D. L. (1993). Artificial neural networks for land-cover classification and mapping. *International Journal of Geographical Information Science*, 7(2) :173–186.

[Corbière et al., 2019] Corbière, C., Thome, N., Bar-Hen, A., Cord, M., and Pérez, P. (2019). Addressing failure prediction by learning model confidence. *Advances in Neural Information Processing Systems*, 32 :2902–2913.

[Cordts et al., 2016] Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. (2016). The cityscapes dataset for semantic urban scene understanding. In *Conference on Computer Vision and Pattern Recognition*. IEEE.

[Dang et al., 2019] Dang, S., Cao, Z., Cui, Z., Pi, Y., and Liu, N. (2019). Open set incremental learning for automatic target recognition. *IEEE Transactions on Geoscience and Remote Sensing*, 57(7) :4445–4456.

[Dasgupta and Hsu, 2008] Dasgupta, S. and Hsu, D. (2008). Hierarchical sampling for active learning. In *International Conference on Machine learning*, pages 208–215. PMLR.

[Demir and Bruzzone, 2014] Demir, B. and Bruzzone, L. (2014). A novel active learning method in relevance feedback for content-based remote sensing image retrieval. *IEEE Transactions on Geoscience and Remote Sensing*, 53(5) :2323–2334.

[Demir et al., 2010] Demir, B., Persello, C., and Bruzzone, L. (2010). Batch-mode active-learning methods for the interactive classification of remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 49(3) :1014–1031.

[Deng et al., 2009] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). ImageNet : A large-scale hierarchical image database. In *Conference on Computer Vision and Pattern Recognition*, pages 248–255. IEEE.

[Derpanis et al., 2012] Derpanis, K. G., Sizintsev, M., Cannons, K. J., and Wildes, R. P. (2012). Action spotting and recognition based on a spatiotemporal orientation analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(3) :527–540.

[Devlin et al., 2019] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT : Pre-training of deep bidirectional transformers for language understanding. In *Conference of the North American Chapter of the Association for Computational Linguistics*, pages 4171–4186. ACL.

[DeVries and Taylor, 2018] DeVries, T. and Taylor, G. W. (2018). Learning confidence for out-of-distribution detection in neural networks. *arXiv preprint arXiv :1802.04865*.

[Ding et al., 2020] Ding, H., Cohen, S., Price, B., and Jiang, X. (2020). Phraseclick : Toward achieving flexible interactive segmentation by phrase and click. In *European Conference on Computer Vision*, pages 417–435. Springer.

[dos Santos et al., 2013] dos Santos, J. A., Gosselin, P., Philipp-Foliguet, S., Torres, R. d. S., and Falcão, A. X. (2013). Interactive multiscale classification of high-resolution remote sensing images. *Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 6, no. 4(4) :2020–2034.

[Dosovitskiy et al., 2021] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. (2021). An image is worth 16x16 words : Transformers for image recognition at scale. *International Conference on Learning Representations*.

[Douillard et al., 2020] Douillard, A., Cord, M., Ollion, C., Robert, T., and Valle, E. (2020). Podnet : Pooled outputs distillation for small-tasks incremental learning. In *European Conference on Computer Vision*, pages 86–102. Springer.

[Dumas et al., 2022] Dumas, L., Defonte, V., Steux, Y., and Sarrazin, E. (2022). Improving pairwise DSM with 3SGM : A semantic segmentation for SGM using an automatically refined neural network. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2 :167–175.

[Dumoulin and Visin, 2016] Dumoulin, V. and Visin, F. (2016). A guide to convolution arithmetic for deep learning. *arXiv preprint arXiv :1603.07285*.

[Everingham et al., 2010] Everingham, M., Van Gool, L., Williams, C. K., Winn, J., and Zisserman, A. (2010). The PASCAL Visual Object Classes (VOC) challenge. *International Journal of Computer Vision*, 88(2) :303–338.

[Fang et al., 2017] Fang, M., Li, Y., and Cohn, T. (2017). Learning how to active learn : A deep reinforcement learning approach. In *Conference on Empirical Methods in Natural Language Processing*, pages 595–605. ACL.

[Ferecatu and Boujemaa, 2007] Ferecatu, M. and Boujemaa, N. (2007). Interactive remote-sensing image retrieval using active relevance feedback. *IEEE Transactions on Geoscience and Remote Sensing*, 45(4) :818–826.

[French et al., 2020] French, G., Laine, S., Aila, T., Mackiewicz, M., and Finlayson, G. (2020). Semi-supervised semantic segmentation needs strong, varied perturbations. In *British Machine Vision Conference*. BMVA.

[Gal and Ghahramani, 2016] 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. IEEE.

[Gal et al., 2017] Gal, Y., Islam, R., and Ghahramani, Z. (2017). Deep bayesian active learning with image data. In *International Conference on Machine Learning*, pages 1183–1192. PMLR.

[Garcia-Garcia et al., 2018] Garcia-Garcia, A., Orts-Escolano, S., Oprea, S., Villena-Martinez, V., Martinez-Gonzalez, P., and Garcia-Rodriguez, J. (2018). A survey on deep learning techniques for image and video semantic segmentation. *Applied Soft Computing*, 70 :41–65.

[García Rodríguez et al., 2020] García Rodríguez, C., Vitrià, J., and Mora, O. (2020). Uncertainty-based human-in-the-loop deep learning for land cover segmentation. *Remote Sensing*, 12(22) :3836.

[Gidaris et al., 2018] Gidaris, S., Singh, P., and Komodakis, N. (2018). Unsupervised representation learning by predicting image rotations. In *International Conference on Learning Representations*. IEEE.

[Guo et al., 2017] Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. (2017). On calibration of modern neural networks. In *International Conference on Machine Learning*, pages 1321–1330. IEEE.

[Hansen and Salamon, 1990] Hansen, L. K. and Salamon, P. (1990). Neural network ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(10) :993–1001.

[He et al., 2017] He, K., Gkioxari, G., Dollar, P., and Girshick, R. (2017). Mask R-CNN. In *International Conference on Computer Vision*, pages 2961–2969. IEEE.

[He et al., 2010] He, K., Sun, J., and Tang, X. (2010). Guided image filtering. In *European Conference on Computer Vision*, pages 1–14. Springer.

[He et al., 2015] He, K., Zhang, X., Ren, S., and Sun, J. (2015). Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(9) :1904–1916.

[He et al., 2016] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Conference on Computer Vision and Pattern Recognition*, pages 770–778. IEEE.

[Hendrycks and Gimpel, 2017] Hendrycks, D. and Gimpel, K. (2017). A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations*. IEEE.

[Hoffman et al., 2018] Hoffman, J., Tzeng, E., Park, T., Zhu, J.-Y., Isola, P., Saenko, K., Efros, A., and Darrell, T. (2018). CyCADA : Cycle-consistent adversarial domain adaptation. In *International Conference on Machine Learning*, pages 1989–1998. PMLR.

[Hu et al., 2019] Hu, Y., Soltoggio, A., Lock, R., and Carter, S. (2019). A fully convolutional two-stream fusion network for interactive image segmentation. In *Neural Networks*, volume 109, pages 31–42. Elsevier.

[Hua et al., 2021] Hua, Y., Marcos, D., Mou, L., Zhu, X. X., and Tuia, D. (2021). Semantic segmentation of remote sensing images with sparse annotations. *IEEE Geoscience and Remote Sensing Letters*.

[Inoue et al., 1993] Inoue, A., Fukue, K., Shimoda, H., and Sakata, T. (1993). A classification method using spatial information extracted by neural network. In *International Geoscience and Remote Sensing Symposium*, pages 893–895. IEEE.

[Ioffe and Szegedy, 2015] Ioffe, S. and Szegedy, C. (2015). Batch normalization : Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, pages 448–456. PMLR.

[Jaegle et al., 2022] Jaegle, A., Borgeaud, S., Alayrac, J.-B., Doersch, C., Ionescu, C., Ding, D., Koppula, S., Zoran, D., Brock, A., Shelhamer, E., et al. (2022). Perceiver IO : A general architecture for structured inputs & outputs. In *International Conference on Learning Representations*. IEEE.

[Jang and Kim, 2019] Jang, W.-D. and Kim, C.-S. (2019). Interactive image segmentation via backpropagating refinement scheme. In *Conference on Computer Vision and Pattern Recognition*, pages 5297–5306. IEEE.

[Kampffmeyer et al., 2016] Kampffmeyer, M., Salberg, A.-B., and Jenssen, R. (2016). Semantic segmentation of small objects and modeling of uncertainty in urban remote sensing images using deep convolutional neural networks. In *Conference on Computer Vision and Pattern Recognition workshops*, pages 1–9. IEEE.

[Kellenberger et al., 2019] 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.

[Kellenberger et al., 2021] Kellenberger, B., Tasar, O., Bhushan Damodaran, B., Courty, N., and Tuia, D. (2021). Deep domain adaptation in Earth observation. *Deep Learning for the Earth Sciences : A Comprehensive Approach to Remote Sensing, Climate Science, and Geosciences*, pages 90–104.

[Kemker et al., 2018] Kemker, R., Salvaggio, C., and Kanan, C. (2018). Algorithms for semantic segmentation of multispectral remote sensing imagery using deep learning. *ISPRS Journal of Photogrammetry and Remote Sensing*, 145 :60–77.

[Kendall and Gal, 2017] Kendall, A. and Gal, Y. (2017). What uncertainties do we need in bayesian deep learning for computer vision ? *Advances in Neural Information Processing Systems*, 30.

[Kirkpatrick et al., 2017] Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A. A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., et al. (2017). Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13) :3521–3526.

[Kontogianni et al., 2020] Kontogianni, T., Gygli, M., Uijlings, J., and Ferrari, V. (2020). Continuous adaptation for interactive object segmentation by learning from corrections. In *European Conference on Computer Vision*, pages 579–596. Springer.

[Krizhevsky et al., 2012] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 25 :1097–1105.

[Laroze et al., 2018] Laroze, M., Dambreville, R., Friguet, C., Kijak, E., and Lefèvre, S. (2018). Active learning to assist annotation of aerial images in environmental surveys. In *International Conference on Content-Based Multimedia Indexing*, pages 1–6. IEEE.

[Le Saux, 2014] Le Saux, B. (2014). Interactive design of object classifiers in remote sensing. In *International Conference on Pattern Recognition*, pages 2572–2577. IEEE.

[LeCun et al., 1989] LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4) :541–551.

[Lee et al., 2018] Lee, K., Lee, K., Lee, H., and Shin, J. (2018). A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in Neural Information Processing Systems*, 31 :7167–7177.

[Lewis and Catlett, 1994] Lewis, D. D. and Catlett, J. (1994). Heterogeneous uncertainty sampling for supervised learning. In *Machine Learning*, pages 148–156. Elsevier.

[Li et al., 2021] Li, Z., Zhang, X., Xiao, P., and Zheng, Z. (2021). On the effectiveness of weakly supervised semantic segmentation for building extraction from high-resolution remote sensing imagery. *Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14 :3266–3281.

[Liang et al., 2018] Liang, S., Li, Y., and Srikant, R. (2018). Principled detection of out-of-distribution examples in neural networks. In *International Conference on Learning Representations*. IEEE.

[Liew et al., 2017] Liew, J., Wei, Y., Xiong, W., Ong, S.-H., and Feng, J. (2017). Regional interactive image segmentation networks. In *International Conference on Computer Vision*, pages 2746–2754. IEEE.

[Liew et al., 2021] Liew, J. H., Cohen, S., Price, B., Mai, L., and Feng, J. (2021). Deep interactive thin object selection. In *Winter Conference on Applications of Computer Vision*, pages 305–314. IEEE.

[Liew et al., 2019] Liew, J. H., Cohen, S., Price, B., Mai, L., Ong, S.-H., and Feng, J. (2019). MultiSeg : Semantically meaningful, scale-diverse segmentations from minimal user input. In *International Conference on Computer Vision*, pages 662–670. IEEE.

[Lin et al., 2017] Lin, H., Shi, Z., and Zou, Z. (2017). Maritime semantic labeling of optical remote sensing images with multi-scale fully convolutional network. *Remote sensing*, 9(5) :480.

[Lin et al., 2020] Lin, Z., Zhang, Z., Chen, L.-Z., Cheng, M.-M., and Lu, S.-P. (2020). Interactive image segmentation with first click attention. In *Conference on Computer Vision and Pattern Recognition*, pages 13339–13348. IEEE.

[Ling et al., 2019] Ling, H., Gao, J., Kar, A., Chen, W., and Fidler, S. (2019). Fast interactive object annotation with Curve-GCN. In *Conference on Computer Vision and Pattern Recognition*, pages 5257–5266. IEEE.

[Liu et al., 2021] Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. (2021). Swin transformer : Hierarchical vision transformer using shifted windows. In *International Conference on Computer Vision*, pages 10012–10022. IEEE.

[Liu et al., 2019] Liu, Z., Wang, J., Gong, S., Lu, H., and Tao, D. (2019). Deep reinforcement active learning for human-in-the-loop person re-identification. In *International Conference on Computer Vision*, pages 6122–6131. IEEE.

[Long et al., 2015] Long, J., Shelhamer, E., and Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Conference on Computer Vision and Pattern Recognition*, pages 3431–3440. IEEE.

[Lowe, 2004] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2) :91–110.

[Lucas et al., 2021] Lucas, B., Pelletier, C., Schmidt, D., Webb, G. I., and Petitjean, F. (2021). A bayesian-inspired, deep learning-based, semi-supervised domain adaptation technique for land cover mapping. *Machine Learning*, pages 1–33.

[Maggiori et al., 2017] Maggiori, E., Tarabalka, Y., Charpiat, G., and Alliez, P. (2017). Can semantic labeling methods generalize to any city ? the INRIA aerial image labeling benchmark. In *International Geoscience and Remote Sensing Symposium*, pages 3226–3229. IEEE.

[Mahadevan et al., 2018] Mahadevan, S., Voigtlaender, P., and Leibe, B. (2018). Iteratively trained interactive segmentation. In *British Machine Vision Conference*. BMVA.

[Maninis et al., 2018] Maninis, K.-K., Caelles, S., Pont-Tuset, J., and Van Gool, L. (2018). Deep Extreme Cut : from extreme points to object segmentation. In *Conference on Computer Vision and Pattern Recognition*, pages 616–625. IEEE.

[Michieli and Zanuttigh, 2021] Michieli, U. and Zanuttigh, P. (2021). Continual semantic segmentation via repulsion-attraction of sparse and disentangled latent representations. In *Conference on Computer Vision and Pattern Recognition*, pages 1114–1124. IEEE.

[Moon et al., 2020] Moon, J., Kim, J., Shin, Y., and Hwang, S. (2020). Confidence-aware learning for deep neural networks. In *International Conference on Machine Learning*, pages 7034–7044. PMLR.

[Nieuwenhuis and Cremers, 2012] Nieuwenhuis, C. and Cremers, D. (2012). Spatially varying color distributions for interactive multilabel segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(05) :1234–1247.

[Nieuwenhuis et al., 2014] Nieuwenhuis, C., Hawe, S., Kleinsteuber, M., and Cremers, D. (2014). Co-sparse textural similarity for interactive segmentation. In *European Conference on Computer Vision*, pages 285–301. Springer.

[Noroozi and Favaro, 2016] Noroozi, M. and Favaro, P. (2016). Unsupervised learning of visual representations by solving jigsaw puzzles. In *European Conference on Computer Vision*, pages 69–84. Springer.

[Ouali et al., 2020] Ouali, Y., Hudelot, C., and Tami, M. (2020). Semi-supervised semantic segmentation with cross-consistency training. In *Conference on Computer Vision and Pattern Recognition*, pages 12674–12684. IEEE.

[Pan and Yang, 2009] Pan, S. J. and Yang, Q. (2009). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10) :1345–1359.

[Panareda Busto and Gall, 2017] Panareda Busto, P. and Gall, J. (2017). Open set domain adaptation. In *International Conference on Computer Vision*, pages 754–763. IEEE.

[Pathak et al., 2016] Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., and Efros, A. A. (2016). Context encoders : Feature learning by inpainting. In *Conference on Computer Vision and Pattern Recognition*, pages 2536–2544. IEEE.

[Ramesh et al., 2021] Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., and Sutskever, I. (2021). Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR.

[Rebuffi et al., 2017] Rebuffi, S.-A., Kolesnikov, A., Sperl, G., and Lampert, C. H. (2017). iCaRL : Incremental classifier and representation learning. In *Conference on Computer Vision and Pattern Recognition*, pages 2001–2010. IEEE.

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

[Roli and Fumera, 2001] Roli, F. and Fumera, G. (2001). Support vector machines for remote sensing image classification. In *Image and Signal Processing for Remote Sensing VI*, volume 4170, pages 160–166. International Society for Optics and Photonics.

[Romera et al., 2017] Romera, E., Alvarez, J. M., Bergasa, L. M., and Arroyo, R. (2017). ERFNet : Efficient residual factorized convnet for real-time semantic segmentation. *IEEE Transactions on Intelligent Transportation Systems*, 19(01) :263–272.

[Ronneberger et al., 2015] Ronneberger, O., Fischer, P., and Brox, T. (2015). U-Net : Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer.

[Roscher et al., 2020] Roscher, R., Volpi, M., Mallet, C., Drees, L., and Wegner, J. D. (2020). SemCity Toulouse : A benchmark for building instance segmentation in satellite images. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 5 :109–116.

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

[Rother et al., 2004] Rother, C., Kolmogorov, V., and Blake, A. (2004). GrabCut : Interactive foreground extraction using iterated graph cuts. *ACM Transactions On Graphics*, 23, no. 3(3) :309–314.

[Rottensteiner et al., 2012] Rottensteiner, F., Sohn, G., Jung, J., Gerke, M., Baillard, C., Benitez, S., and Breitkopf, U. (2012). The ISPRS benchmark on urban object classification and 3D building reconstruction. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 1(01) :293–298.

[Rumelhart et al., 1986] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088) :533–536.

[Ružička et al., 2020] Ružička, V., D'Aronco, S., Wegner, J. D., and Schindler, K. (2020). Deep active learning in remote sensing for data efficient change detection. In *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases MACLEAN Workshop*. Springer.

[Saffari et al., 2009] Saffari, A., Leistner, C., Santner, J., Godec, M., and Bischof, H. (2009). On-line random forests. In *International Conference on Computer Vision*, pages 1393–1400. IEEE.

[Santner et al., 2010] Santner, J., Pock, T., and Bischof, H. (2010). Interactive multi-label segmentation. In *Asian Conference on Computer Vision*, pages 397–410. Springer.

[Schmitt et al., 2021] Schmitt, M., Ahmadi, S. A., and Hänsch, R. (2021). There is no data like more data-current status of machine learning datasets in remote sensing. In *International Geoscience and Remote Sensing Symposium*, pages 1206–1209. IEEE.

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

[Settles, 2009] Settles, B. (2009). Active learning literature survey. Technical report, University of Wisconsin-Madison Department of Computer Sciences.

[Shannon, 1948] Shannon, C. E. (1948). A mathematical theory of communication. *The Bell system technical journal*, 27(3) :379–423.

[Shi and Malik, 2000] Shi, J. and Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8) :888–905.

[Shotton et al., 2009] Shotton, J., Winn, J., Rother, C., and Criminisi, A. (2009). Textonboost for image understanding : Multi-class object recognition and segmentation by jointly modeling texture, layout, and context. *International journal of computer vision*, 81(1) :2–23.

[Sofiiuk et al., 2020] Sofiiuk, K., Petrov, I., Barinova, O., and Konushin, A. (2020). f-BRS : Rethinking backpropagating refinement for interactive segmentation. In *Conference on Computer Vision and Pattern Recognition*, pages 8623–8632. IEEE.

[Sohn et al., 2020] Sohn, K., Berthelot, D., Carlini, N., Zhang, Z., Zhang, H., Raffel, C. A., Cubuk, E. D., Kurakin, A., and Li, C.-L. (2020). Fixmatch : Simplifying semi-supervised learning with consistency and confidence. *Advances in Neural Information Processing Systems*, 33 :596–608.

[Souly et al., 2017] Souly, N., Spampinato, C., and Shah, M. (2017). Semi supervised semantic segmentation using generative adversarial network. In *International Conference on Computer Vision*, pages 5688–5696. IEEE.

[Srivastava et al., 2014] 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(1) :1929–1958.

[Strudel et al., 2021] Strudel, R., Garcia, R., Laptev, I., and Schmid, C. (2021). Segmenter : Transformer for semantic segmentation. In *International Conference on Computer Vision*, pages 7262–7272. IEEE.

[Sunkara et al., 2020] Sunkara, V., Purri, M., Le Saux, B., and Adams, J. (2020). Street to cloud : Improving flood maps with crowdsourcing and semantic segmentation. In *Tackling Climate Change with Machine Learning workshop at NeurIPS*.

[Tan et al., 2018] Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., and Liu, C. (2018). A survey on deep transfer learning. In *International Conference on Artificial Neural Networks*, pages 270–279. Springer.

[Tan and Le, 2019] Tan, M. and Le, Q. (2019). EfficientNet : Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*, pages 6105–6114. PMLR.

[Tarvainen and Valpola, 2017] Tarvainen, A. and Valpola, H. (2017). Mean teachers are better role models : Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in Neural Information Processing Systems*, 30.

[Tasar et al., 2020a] Tasar, O., Giros, A., Tarabalka, Y., Alliez, P., and Clerc, S. (2020a). DAugNet : Unsupervised, multisource, multitarget, and life-long domain adaptation for semantic segmentation of satellite images. *IEEE Transactions on Geoscience and Remote Sensing*, 59(2) :1067–1081.

[Tasar et al., 2019] Tasar, O., Tarabalka, Y., and Alliez, P. (2019). Incremental learning for semantic segmentation of large-scale remote sensing data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(9) :3524–3537.

[Tasar et al., 2020b] Tasar, O., Tarabalka, Y., Giros, A., Alliez, P., and Clerc, S. (2020b). StandardGAN : Multi-source domain adaptation for semantic segmentation of very high resolution satellite images by data standardization. In *Conference on Computer Vision and Pattern Recognition Workshops*, pages 192–193. IEEE.

[Tuia et al., 2011a] Tuia, D., Pasolli, E., and Emery, W. J. (2011a). Using active learning to adapt remote sensing image classifiers. *Remote Sensing of Environment*, 115(9) :2232–2242.

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

[Vaswani et al., 2017] 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.

[Vaudaux-Ruth et al., 2021] Vaudaux-Ruth, G., Chan-Hon-Tong, A., and Achard, C. (2021). ActionSpotter : Deep reinforcement learning framework for temporal action spotting in videos. In *International Conference on Pattern Recognition*, pages 631–638. IEEE.

[Vincent and Soille, 1991] Vincent, L. and Soille, P. (1991). Watersheds in digital spaces : an efficient algorithm based on immersion simulations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(06) :583–598.

[Wang et al., 2019a] Wang, Y., Zhou, Q., Liu, J., Xiong, J., Gao, G., Wu, X., and Latecki, L. J. (2019a). LEDNet : A lightweight encoder-decoder network for real-time semantic segmentation. In *International Conference on Image Processing*, pages 1860–1864. IEEE.

[Wang et al., 2019b] Wang, Z., Acuna, D., Ling, H., Kar, A., and Fidler, S. (2019b). Object instance annotation with deep extreme level set evolution. In *Conference on Computer Vision and Pattern Recognition*, pages 7500–7508. IEEE.

[Weiss et al., 2016] Weiss, K., Khoshgoftaar, T. M., and Wang, D. (2016). A survey of transfer learning. *Journal of Big Data*, 3(1) :1–40.

[Xie et al., 2021] Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J. M., and Luo, P. (2021). SegFormer : Simple and efficient design for semantic segmentation with transformers. *Advances in Neural Information Processing Systems*, 34.

[Xu et al., 2016] Xu, N., Price, B., Cohen, S., Yang, J., and Huang, T. (2016). Deep interactive object selection. In *Conference on Computer Vision and Pattern Recognition*, pages 373–381. IEEE.

[Yao et al., 2012] Yao, A., Gall, J., Leistner, C., and Van Gool, L. (2012). Interactive object detection. In *Conference on Computer Vision and Pattern Recognition*, pages 3242–3249. IEEE.

[Yoo and Kweon, 2019] Yoo, D. and Kweon, I. S. (2019). Learning loss for active learning. In *Conference on Computer Vision and Pattern Recognition*, pages 93–102. IEEE.

[Yu and Koltun, 2016] Yu, F. and Koltun, V. (2016). Multi-scale context aggregation by dilated convolutions. In *International Conference on Learning Representations*. IEEE.

[Yuan et al., 2021] Yuan, X., Shi, J., and Gu, L. (2021). A review of deep learning methods for semantic segmentation of remote sensing imagery. *Expert Systems with Applications*, 169 :114417.

[Yue et al., 2019] Yue, K., Yang, L., Li, R., Hu, W., Zhang, F., and Li, W. (2019). TreeUNet : Adaptive tree convolutional neural networks for subdecimeter aerial image segmentation. *ISPRS Journal of Photogrammetry and Remote Sensing*, 156 :1–13.

[Zhang et al., 2020] Zhang, S., Liew, J. H., Wei, Y., Wei, S., and Zhao, Y. (2020). Interactive object segmentation with inside-outside guidance. In *Conference on Computer Vision and Pattern Recognition*, pages 12234–12244. IEEE.

[Zhang et al., 2021] Zhang, Y., Liu, H., and Hu, Q. (2021). TransFuse : Fusing Transformers and CNNs for Medical Image Segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 14–24. Springer.

[Zheng et al., 2021] 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 *Conference on Computer Vision and Pattern Recognition*, pages 6881–6890. IEEE.

[Zhuang et al., 2020] Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., Xiong, H., and He, Q. (2020). A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 109(1) :43–76.

Appendix A - Résumé en français

A.1. Introduction

La segmentation sémantique, c'est-à-dire la classification d'une image au niveau du pixel, est extrêmement importante en télédétection et est traitée par des réseaux de neurones profonds pour une variété d'applications telles que la cartographie de la couverture terrestre, la détection des changements ou la surveillance des terres agricoles. Cette tâche est intrinsèquement complexe et, bien que les réseaux de neurones profonds puissent être très efficaces, ils sont toujours sujets à l'échec. En effet, même sur des benchmarks académiques, les méthodes actuelles de l'état de l'art nécessitent souvent des architectures spécifiques et un réglage fin pour obtenir des performances élevées mais des résultats encore imparfaits. De plus, les réseaux de neurones profonds sont particulièrement sensibles à différents facteurs tels que l'adaptation du domaine entre les données d'entraînement et de test inhérente aux données de télédétection (différents temps, zones géographiques, types de capteurs, ombres des nuages, etc.) ou la difficulté d'avoir accès à de grands jeux de données annotés pour chaque application commerciale spécifique, même si de nombreux efforts sont faits par la communauté dans ce sens. Ainsi, l'incertitude sur la qualité des résultats des réseaux neuronaux rend leur déploiement compliqué pour de nombreux cas d'usages industriels.

L'apprentissage interactif (IL) est une solution possible à ce problème. Il s'agit d'ajouter un humain dans la boucle pour travailler en synergie avec un algorithme d'apprentissage afin de l'adapter aux entrées de l'utilisateur. Dans cette thèse, les interactions avec l'utilisateur prennent la forme de points cliqués et représentent les labels de segmentation, choisis par l'utilisateur.

Dans cette thèse, nous explorons l'IL pour la segmentation sémantique sur des images de télédétection. Plus précisément, nous cherchons à répondre à trois questions de recherche principale :

— Comment interagir avec les réseaux de neurones après l'apprentissage ?
  En général, les réseaux de neurones sont formés hors ligne et sont ensuite utilisés de manière entièrement automatique. Nous étudions comment l'interactivité après l'entraînement peut permettre d'affiner les prédictions de ces algorithmes.
— Comment obtenir des données pertinentes ?
  Les annotations de l'utilisateur sont par nature sparses par rapport aux images, et cela est encore plus vrai en télédétection où les images peuvent être énormes. Il peut également être particulièrement fastidieux pour un utilisateur d'examiner en détail chaque carte de segmentation. Afin d'alléger la charge de travail de l'annotateur, nous étudions comment guider l'annotateur à travers l'image en sélectionnant les données pertinentes à

annoter.

— How to adapt to new data and new use cases ?

Le processus standard en Intelligence Artificielle pour l'Observation de la Terre consiste à apprendre sur un ensemble de données pré-annotées pour produire un modèle statique. Cependant, ce processus est plutôt adapté à des tâches bien définies dans des environnements fermés. En effet, bien que de tels modèles puissent bien fonctionner sur les benchmarks, ils vont souvent faire des erreurs dans les environnements ouverts et dynamiques de la vie réelle. En outre, il peut également être nécessaire de prévoir de nouvelles classes de segmentation cibles et donc d'adapter l'espace de sortie des algorithmes.

Pour répondre à ces problèmes, nous présentons un framework général d'apprentissage interactif pour la segmentation sémantique d'images de télédétection dont les composantes sont détaillées dans les différents chapitres de cette thèse.

A.2. Revue de littérature

Ce chapitre présente l'état de l'art des outils d'intelligence artificielle pertinents pour la compréhension de la thèse. Il commence par donner un bref aperçu de la vision par ordinateur et par rappeler l'origine et les composants des réseaux de neurones convolutifs actuels. Ensuite, il entre dans les détails de l'origine de la segmentation sémantique et présente certaines des architectures de réseaux de neurones actuelles conçues pour y répondre. Enfin, il explique les différentes limites de ces architectures et comment une approche interactive peut y remédier.

Dans un deuxième temps, il traite de l'apprentissage interactif et passe en revue les travaux qui ont inspiré ceux de cette thèse ou qui traitent de problèmes similaires. Il présente d'abord un panel de travaux d'interprétation interactive en télédétection. Il détaille ensuite les algorithmes historiques et les algorithmes récents basés sur l'apprentissage profond conçus pour la segmentation interactive. Il décrit enfin les stratégies d'apprentissage actif et se concentre spécifiquement sur les méthodologies basées sur l'incertitude.

Ce chapitre se conclue par une présentation des jeux de données publics et des métriques utilisées pour évaluer les différents travaux effectués pendant la thèse. Il y est également souligné les limites des réseaux de neurones profonds dans la segmentation sémantique à travers des exemples tirés de ces jeux de données.

A.3. Apprentissage interactif rapide

Ce chapitre propose l'algorithme DISIR dans un cadre interactif de segmentation multi-classes pour les images aériennes. En partant d'un réseau de neurones conçu pour la segmentation sémantique, il s'agit d'entraîner ce réseau à la fois classique-

ment pour la segmentation et à exploiter les annotations de l'utilisateur pour se laisser guider dans sa tâche. Les annotations utilisateurs modifient seulement les entrées du réseau de neurones et pas ses poids. Au moment du test, les annotations de l'utilisateur sont donc données au réseau sans modifier ses paramètres, ce qui rend le processus de segmentation sémantique interactif rapide et efficace.

Grâce à des expériences sur des jeux de données de télédétection publics, il est montré que le raffinement interactif est efficace pour toutes les classes de segmentations dans des jeux de données allant de deux à six classes de segmentation. Il améliore les résultats de classification de 4% en moyenne pour 120 clics et, surtout, produit des cartes de segmentation visuellement améliorées. Il y est aussi montré que DISIR est efficace quelle que soit l'architecture du réseau. Différentes représentations des annotations sont étudiées et il est conclu que les clics positionnés à l'intérieur des objets et encodés à l'aide de la transformée de distance sont les plus porteurs d'information, même si la stratégie d'encodage ne modifie pas beaucoup les résultats. En effet, le réseau est capable d'appréhender le contexte de l'annotation à partir de l'annotation et de l'image seules, sans filtre supplémentaire sur l'encodage de l'annotation. Cependant, si la résolution spatiale n'a pas d'impact sur l'efficacité de la méthode proposée, le manque de données d'entraînement empêche les réseaux de neurones d'apprendre à être guidés par les annotations. Ceci est particulièrement un problème pour le jeu de données SemCity Toulouse qui contient relativement peu de données annotées.

L'autre limite principale identifiée d'un réseau entraîné avec DISIR est la propagation spatiale des informations fournies par les annotations, car il n'est pas capable de propager ces informations loin de l'annotation correspondante.

A.4. Apprentissage interactif à l'échelle

Ce chapitre tente justement de proposer des solutions pour davantage propager les annotations. Deux méthodes sont proposées pour résoudre ce problème. La première, issue des architectures Transformer utilisées en Traitement Automatique du Langage, ajoute un mécanisme d'attention pour propager l'information des annotations au delà des champs réceptifs des noyaux de convolution. Cependant, en plus d'être coûteuse en temps de calcul, cette solution obtient des résultats mitigés et n'améliore pas les résultats autant que simplement augmenter la taille des champs réceptifs des noyaux de convolution.

La deuxième proposition, appelée DISCA, repose sur DISIR et se base sur le ré-entraînement à la volée du réseau de neurones en considérant les annotations utilisateurs comme des cartes de vérité terrain sparses. Afin de ne pas faire d'oubli catastrophique en sur-apprenant les annotations, un terme de régularisation utilisant la prédiction initiale du réseau est ajoutée afin de stabiliser les prédictions. Cette stratégie d'entraînement interactif est complémentaire à DISIR. En effet, ayant un coût de calcul plus élevé dû au réapprentissage à la volée, DISCA est adapté pour

corriger des erreurs relativement importantes, comme cela va être le cas dans de l'adaptation de domaines par exemple. DISIR, étant plus rapide mais plus local, va davantage être adapté pour corriger des erreurs plus localement.

A.5. Guider les interactions

Les images de télédétection peuvent être extrêmement volumineuses et il peut alors être fastidieux pour un opérateur de les passer entièrement en revue. Pour résoudre ce problème, ce chapitre propose une amélioration méthodologique reposant sur l'apprentissage actif et l'estimation d'incertitude des réseaux de neurones pour guider rapidement l'utilisateur vers des requêtes représentant les zones les plus significatives des images à annoter. Différentes fonctions d'acquisition pour estimer l'incertitude du réseau sont analysées et il est conclu que celle se basant sur le calcul de l'entropie en sortie de réseau ainsi que celle utilisant un réseau secondaire pour prédire la confiance du premier semblent être les plus adaptées pour ce cas d'usage. Globalement, guider l'utilisateur à partir de l'estimation d'incertitude permet d'atteindre plus rapidement des performances élevées, et permet ainsi de réduire le nombre d'interactions pour atteindre une précision de classification donnée.

A.6. Vers de la segmentation interactive incrémentale

Ce chapitre s'attaque à l'apprentissage de nouvelles classes de segmentation de façon interactive. S'appuyant sur le schéma de DISCA considérant les annotations utilisateurs comme des points de vérité terrain, sur des régularisations pertinentes et sur du *pseudo-labelling*, il est montré qu'il est possible d'apprendre une nouvelle classe dans un tel contexte. Notamment, la régularisation SDR, s'appuyant sur une disposition en espace latent, semble particulièrement adaptée à cette tâche, même si elle augmente considérablement le temps de traitement. En effet, non seulement elle améliore constamment les résultats, mais elle stabilise également la convergence des réseaux de neurones dans ce contexte. Cependant, pour utiliser optimalement la méthode proposée, il est nécessaire de rassembler environ une centaine d'annotations utilisateur par image. Ce nombre est prohibitif dans un contexte interactif et des travaux futurs sont prévus pour améliorer cette approche.