POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Learning to Detect Illegal Landfills in Aerial Images with Scarce Labeling Data

Tesi di Laurea Magistrale in
Computer Science and Engineering

Authors: **Corrado Fasana, Samuele Pasini**

Students IDs: 962672, 964046
Advisor: Prof. Piero Fraternali
Co-advisor: Federico Milani
Academic Year: 2021-2022

Ringraziamenti

Eccomi qui. Finalmente sono giunto alla conclusione dei miei studi. Il tempo è sembrato volare via in fretta e allo stesso tempo è sembrato interminabile. Mi sembra doveroso però dedicare un attimo a ringraziare tutte le persone che mi hanno accompagnato durante questa esperienza e che hanno contribuito non solo alla mia formazione accademica, ma anche a dare forma alla mia personalità.
Un ringraziamento speciale va innanzitutto al mio relatore, prof. Piero Fraternali, per la sua disponibilità e tempestività in ogni istante durante la stesura dell'elaborato. Grazie, per aver sempre dimostrato grande passione e dedizione per il suo lavoro e soprattutto interesse per il mio futuro.
Un ringraziamento particolare va al mio co-relatore Federico Milani, per tutto il tempo dedicato ad affiancarmi nella scrittura della tesi. Grazie, per tutti i consigli che hai sempre fornito e per la prontezza nel rispondere a qualunque richiesta.
Un importante ringraziamento va ovviamente a Samuele, co-autore di questa tesi, che ha condiviso con me dal primo all'ultimo momento di Università. Grazie per tutti i momenti di studio che abbiamo passato insieme, per avermi spinto a dare sempre il meglio di me e soprattutto per i bei momenti passati in amicizia anche al di fuori dell'Università. Sono fiero di poter raggiungere questo traguardo insieme.
Nuovamente grazie a Federico, ma anche a Sergio, Nicolò e Rocio per i momenti condivisi negli ultimi due mesi in ufficio. Grazie, per aver reso questo periodo divertente e meno stancante. È stato un piacere conoscervi e poter collaborare con voi in ogni momento.
Un grazie al Politecnico di Milano e a tutti i suoi professori per aver contribuito alla mia formazione e ai miei compagni di corso per i momenti passati insieme a studiare e quelli a giocare a carte prima di ogni lezione.
Un grazie ai miei amici con cui ho condiviso molti bei momenti, soprattutto Francesco, amico e compagno di studi. Grazie a te ho scoperto la bellezza di giocare a pallavolo e di essere parte di una squadra. Per questo, un grazie va anche alla mia squadra di pallavolo per i momenti divertenti e di sfogo che ho passato nell'ultimo anno.
Un grazie davvero sentito e speciale va a Stefania, mia compagna di squadra e mia fidanzata. Grazie per esserci sempre stata in qualunque momento, soprattutto quando ero demoralizzato e stressato. Grazie per essermi sempre stata accanto, per aver creduto nelle mie capacità e per avermi incoraggiato a non farmi mai abbattere quando ero in difficoltà. Grazie per avermi transmesso un po' di quella tua spontaneità e spensieratezza che ogni tanto vale la pena avere. Averti incontrata mi ha reso felice, ma poterti conoscere a fondo per ciò che sei veramente ancora di più.
A questo punto, non mi rimane che ringraziare infinitamente i miei genitori. Grazie per avermi sempre sostenuto e appoggiato in ogni mia decisione, per avermi dato l'opportunità di studiare ciò che mi appassiona senza mai farmi mancare nulla. Grazie per aver sem-

pre creduto nelle mie capacità, per avermi spronato a raggiungere i miei obiettivi e per il supporto che mi avete sempre dimostrato anche nei momenti più difficili. Grazie per aver fatto sì che io diventassi la persona che sono adesso. Probabilmente non potrò mai ringraziarvi abbastanza, ma spero di avervi reso orgogliosi di me.
Un grandissimo grazie anche a mio fratello Simone e mia sorella Eleonora. Grazie per avermi sopportato durante tutta la mia carriera di studi, per aver trovato il tempo di ascoltare i miei dubbi e per avermi sempre dato supporto incodizionatamente.
Grazie ai miei nonni, soprattutto alla mia amata nonna Anna e al mio nonno Bramino. Ho passato la gran parte del tempo a studiare in casa vostra. Siete sempre stati pronti ad ascoltare i racconti interminabili sulle mie avventure universitarie e non. Vi ho sempre raccontato tutto quanto, anche quello che stavo studiando e che per voi poteva essere molto difficile da capire. Eppure, mi avete sempre ascoltato, avete sempre gioito per i traguardi che ho raggiunto e non dimenticherò mai il vostro sorriso e la vostra contentezza nel sentirmi parlare. Non ho avuto la possibiltà di condividere con te, nonno Bramino, la gioia di laurearmi, ma so che se fossi qui saresti fiero di me. A te nonno dedico questa tesi.

Corrado Fasana
Como, 20 dicembre 2022

Vorrei dedicare questo spazio a tutte le persone che mi hanno accompagnato in questo percorso di Tesi.

Grazie al Professor Piero Fraternali, che è stato un punto di riferimento per tutto il mio percorso universitario e, con mio grande piacere, anche il mio relatore. In questi mesi mi ha guidato con grande competenza, disponibilità e puntualità, trasmettendomi inoltre una grande passione per il suo lavoro in grado di spronarmi al continuo miglioramento. Il grande interesse che ha sempre espresso per il mio futuro ha lasciato il segno, e queste poche righe non bastano per esprimere tutta la mia ammirazione.

Vorrei ringraziare il mio co-relatore Federico Milani, che ha enormemente arricchito la qualità di questa Tesi con preziosi consigli, e che mi ha fornito metodi e strumenti in grado di migliorarmi notevolmente dal punto di vista professionale.

Un Grazie enorme al mio co-autore Corrado, con il quale ho condiviso il mio percorso di studi fin dai primi giorni di Università. È stato un piacere lavorare insieme a te su questa Tesi e raggiungere questo traguardo con un grande amico.

Un ringraziamento speciale a Rocio Nahime Torres. Il tuo prezioso lavoro mi ha fornito un solido punto di partenza ed è stato fondamentale per la mia Tesi.

Ringrazio anche Nicolò e Sergio, con i quali ho avuto il piacere di condividere, oltre che con Corrado e Federico, molte giornate in questi mesi, grazie a voi ho potuto apprezzare i benefici di un ambiente di lavoro amichevole e collaborativo.

È doveroso ringraziare i miei genitori, Rosy e Rudy, nonna Rosa e la mia intera famiglia, che ha fatto molti sacrifici per permettermi di proseguire nel mio percorso universitario e mi ha sempre sostenuto.

Un ringraziamento speciale va anche a nonno Peppino. Sin da bambino mi hai insegnato i valori dell'impegno e del lavoro, e avrei tanto voluto condividere questo importante traguardo con te.

Grazie ad Anna, la mia ragazza, che è entrata nella mia vita da diversi anni rendendomi una persona migliore da ogni punto di vista. Sei riuscita nella difficile missione di sopportami durante questi mesi intensi, hai creduto nelle mie capacità anche quando ero io a dubitarne, e sei stata la persona sulla quale ho sempre saputo di poter contare incondizionatamente.

Grazie a tutti i miei amici, che vorrei tanto avere lo spazio di nominare uno ad uno, a partire da quelli che ci sono da sempre e con i quali ho avuto il piacere di crescere, fino a quelli che ho conosciuto nel mio percorso universitario. Mi avete regalato tanti momenti di felicità e spensieratezza, senza i quali sarebbe stato difficile raggiungere questo obiettivo.

Grazie infine al Politecnico di Milano, che mi ha reso la persona che sono oggi.

Samuele Pasini
Como, 20 dicembre 2022

Abstract

Environmental monitoring is essential to understand the conditions of the environment and the changes caused by human activities. The advances in Remote Sensing technologies for earth observation open the possibility of scanning vast territories with the help of satellite imagery. State-of-the-art Deep Learning architectures can be used for this task, but they need fine-grained ground truth annotations built with expert knowledge. To solve this limitation, Self and weakly supervised methods can be used to supplement the lack of manual object-level annotations and pre-trained models, thanks to the abundance of non-annotated images in the remote sensing domain. This work presents a survey of self- and weakly supervised aerial image analysis methods. Then, suitable methods are explored and evaluated on a novel data set (AerialWaste) to identify and localize illegal waste in remote sensing images. The results can help the photo interpretation process currently performed manually by experts in the field.

Keywords: Illegal Landfills detection, Weak Supervision, Self Supervision, Remote Sensing Images

Sommario

Il monitoraggio ambientale è essenziale per comprendere le condizioni dell'ambiente e i cambiamenti causati dalle attività umane. I progressi nelle tecnologie di telerilevamento per l'osservazione della terra aprono la possibilità di scansionare vasti territori con l'aiuto di immagini satellitari. Al fine di raggiungere questo obiettivo, è possibile utilizzare architetture di apprendimento profondo all'avanguardia, che tuttavia, necessitano di annotazioni dettagliate fornite da esperti. Per risolvere questa limitazione, è possibile utilizzare metodi di apprendimento autogestito e debolmente supervisionati per integrare la mancanza di annotazioni manuali a livello di oggetto e di modelli pre-allenati, grazie all'abbondanza di immagini satellitari non annotate. Questo lavoro presenta un'indagine sui metodi di apprendimento autogestito e debolmente supervisionati per immagini satellitari. Metodi adeguati vengono esaminati su un nuovo set di dati (AerialWaste) per identificare e localizzare discariche abusive nelle immagini di telerilevamento. I risultati possono essere di notevole aiuto per il processo di interpretazione fotografica che attualmente viene svolto manualmente da esperti del settore.

Parole chiave: Rilevamento di discariche abusive, Supervisione debole, Apprendimento autogestito, Immagini di telerilevamento

Contents

1 | Introduction

Environmental protection is the practice related to the protection of the natural environment to conserve natural resources and if needed try to repair damages to preserve all the forms of life. Environmental protection is pursued by agencies collaborating all around the globe, with the mission to protect, improve and restore the environment through programs that aim at reducing risks of environmental contamination, which may happen because of the presence of hazardous materials, wastes, fuels, and oils. These programs provide guidelines to avoid pollution, procedures for safely working and managing these materials, as well as actions that should be performed in case pollution cannot be prevented. As a consequence, environmental protection is one of the most challenging and priority missions to be pursued in the world so that a healthy and sustainable future can be guaranteed for all forms of life. This leads to a continuously increasing concern on environment-related topics both by the population and by governments. During the last few years, the importance of these aspects guided nations to define Sustainable Development Goals (SDG) [1].

Aiming to reach the goals of environmental protection, environmental monitoring is fundamental to assess the environment's conditions and its changes due to natural or human interventions. As defined by the United Nations in the 2030 Agenda for Sustainable Development [1], a wide range of topics, such as biodiversity and ecosystems, chemicals and waste, desertification, disaster risk reduction, climate change and water availability, among others, should be considered to reach SDG. Many of these topics are controlled through environmental monitoring activities. All these topics share relevant properties and are sometimes strictly connected with each other, leading to the possibility of developing similar solutions for different problems. However, other factors such as country regulations, geographic characteristics, and availability of the data, force each problem to be faced and studied more in-depth before being sure that the adopted approaches are able to generalize well to different areas.

Among the topics considered to reach SDG, effective control of the generation, storage, treatment, recycling and reuse, transport, recovery and disposal of hazardous wastes are of great importance to ensure proper health, environmental protection, natural resource management, and sustainable development. For this reason, preventing the generation of hazardous wastes is crucial and requires experienced people, financial resources, and technical and scientific knowledge.

When considering this topic, particular attention needs to be paid to illegal landfills. The demographic increase has a considerable impact on the waste generation [2] and this phenomenon could lead to the birth of new unauthorized sites which are a serious source of hazards for the environment and the society [3–5]. Moreover, waste crimes, i.e., activities that violate the waste management laws cover an even more important role in Italy where, according to the 2021 report of Legambiente on Ecomafia [6], 34,867 crimes against the environment and more than 8,000 (assessed) crimes related to waste and landfills were registered in 2020. This phenomenon is even more evident in regions with a more relevant Mafia influence (e.g., Campania, Sicily, Puglia and Lazio). These numbers show a slight increment (+0.6%) in waste-related crimes with respect to the previous year. However, a significant decrease in the monitoring activities (-17%) is also testified.

The main issue is that emissions and toxicological hazards from illegal dump sites can be extremely high compared to regulated landfills [3]. This endangers even more public health given that if waste treatment is not performed carefully, the release of leachate in the environment can pollute water sources and increase cancer incidence in the long term [7].

For this reason, detecting illegal disposal sites on time is crucial to reduce the impacts on both the environment and society. While an on-site inspection of potentially illegal waste disposal sites is still fundamental to assess the danger and potential impacts of illicit activities, it is necessary to reduce the number of locations to be examined. Otherwise, it is not possible to efficiently keep a wide territory under control. To make this process more efficient, it is possible to exploit the availability of Remote Sensing (RS) technologies that allow to capture aerial images and examine them to check the presence or absence of illegal landfills [8]. Being able to distinguish among the different types of objects or storage containers that are present in landfills is even more challenging. This task does not only require correctly identifying the location of a waste disposal site but also capturing relevant aspects that allow distinguishing among the different considered items. Even though experts' interpretation of aerial images is still a predominant technique, the advent of Computer Vision (CV) methods, boosted by the recent advancements in the Deep Learning (DL) field [9, 10], leads the way to the development of new automatic tools that can capture experts knowledge and provide a suitable model for automating, at least partially, the process. This way, a significant amount of data can be processed more rapidly, allowing experts to concentrate their effort on the analysis of only the most relevant areas, reducing the number of on-site inspections, and allowing the coverage of a much wider range of the territory.

An important aspect related to DL methods is the need for huge amounts of data to effectively train a deep architecture. However, these data often require careful annotations depending on the specific task being solved which is a very time-consuming activity [11]. At the same time, the availability of the data itself may be limited by specific regulations, especially in sensitive domains such as waste disposal. For this reason, it is necessary to define new technologies that are able to solve the same tasks by reducing the need for huge quantities of annotated data. In this direction, recent advancements in the research of Weak Supervision (WS) [12, 13] and Self Supervision [14, 15] technologies opens up the possibility of exploiting these approaches for environmental monitoring processes such as illegal landfills detection, reducing the need of carefully annotated data.

In this thesis, the illegal landfills detection problem is considered as a multi-label classification problem, meaning that the focus is on designing a model that is able to differentiate among the different types of landfills exploiting Aerial Images. The main idea behind this approach is that if the classification allows reaching promising performance, then weakly supervised approaches can be applied in the illegal landfills scenario to detect them when only coarse-grained labels are given. The approach illustrated in this thesis exploits a Convolutional Neural Network (CNN) classifier and Remote Sensing Images (RSIs) in the optical range (RGB). Even though DL models were proven effective in many applications, there is still a limited number of approaches that exploit CNN architectures to detect waste, especially in the RS domain. The works [16, 17] exploit CNN fed with Aerial Images to perform waste identification by formulating the problem as an Object Detection (OD) task. In this case, manually crafted Bounding Boxes (BBs) are required to perform OD. Unfortunately, this is very costly and error-prone especially because waste disposal sites are not easily identifiable even in the case in which high-resolution RSIs are used.

To address this issue, the work by Torres et al. [8, 18] proposes to address the task as a scene classification problem, which requires only whole image labels as ground truth, indicating the presence or absence of a landfill. The proposed model reaches 81.9% precision and 79.5% recall on a test set. Unlike previous works, in this thesis, the problem is addressed as a multi-label classification problem, thus, labels indicating the categories of landfills present in each image are needed. These labels can be more costly than those required in [8, 18], but they are still easier to obtain than BBs. To cope with the complexity of illegal landfill imagery, in which objects possess varying scales and appearances, the same multi-scale CNN architecture proposed by Torres et al. [8, 18] is used. The adopted method is tested on a large-scale territory, and both a qualitative and a quantitative evaluation are reported exploiting the ODIN evaluation tool [19, 20]. The proposed method is able to obtain 56.43% average F1-score on the AerialWaste data set for the multi-label classification task while also showing good localization capabilities. In this case, the possibility of generating pseudo-labels to perform Weakly Supervised Instance Segmentation (WSIS) is evaluated. At the moment, the generated segmentations are not good enough to train an instance segmentation network due to the difficulty of the domain under analysis and the need for specific network adaptations.

The contributions of this thesis can be summarized as follows:

- An analysis of the differences between natural images and Remote Sensing Images is conducted given that the identified characteristics have a huge impact on the generalization capabilities of models.
- A summary of State-Of-The-Art (SOTA) Self Supervised and Weakly Supervised (WS) approaches for the detection and segmentation of objects in natural images and RSIs is proposed, with a particular focus on Weakly Supervised Object Detection (WSOD) in RSIs, for which classification and analysis of the main approaches are collected in a published survey [13].
- A summary of the SOTA for landfills detection is provided to define the starting point of the approach proposed in this thesis.
- An analysis of the used AerialWaste data set [18] is provided, highlighting the characteristics and related issues that can potentially hurt the discriminative capabilities of a classifier.
- Different techniques that can mitigate the previously identified issues are proposed to enhance the discriminative power of the classifiers. Several multi-scale multi-label CNN classifiers for the classification task are compared quantitatively, verifying the impact of the proposed techniques.
- The output of the classifiers is evaluated qualitatively by exploiting Class Attention Maps (CAMs) as a visual understanding and interpretability technique. This procedure allows to identify the image regions where a classifier focuses its attention and allows to understand how much the classifier is able to distinguish the different classes.
- The final architecture exploits a ResNet50 backbone [21] augmented with a Feature Pyramid Network (FPN) [22] trained on a data set enlarged with synthetic data and allows to obtain 58.08% average precision, 61.16% average recall and 56.43%

average F1-score on the AerialWaste test set.

- The possibility of generating pseudo-labels to train an instance segmentation network is analyzed. At the moment, the segmentations obtained from the best classification model are not sufficiently good to be used as pseudo-labels.

The rest of the thesis is organized as follows:

Chapter 2 surveys the background about the different Weakly Supervised learning techniques and Self Supervised techniques as well as the background concerning the illegal landfills detection task.

Chapter 3 presents the data set, the architecture and the various methods employed in this thesis to address the problem of illegal landfills detection as a multi-label classification task.

Chapter 4 describes the performed experiments, and the analyses that are conducted on each of them from both a quantitative and a qualitative viewpoint, and the considerations that lead to the selection of the best model.

Chapter 5 draws the conclusions and identifies possible future directions.

2 | Background

This thesis pursues the application of Artificial intelligence (AI) to Remote Sensing Images (RSIs) for Environmental Monitoring purposes, particularly in the context of illegal landfills detection. This chapter introduces the relevant concepts related to this problem and provides an overview of the relevant scientific literature.

The chapter is structured as follows: Section 2.1 introduces RSIs and explains peculiarities and differences with respect to natural images. Section 2.2 introduces the concept of Full Supervision (FS) in the area of Machine Learning (ML) and the needs that lead to the use of Weak Supervision (WS) and Self Supervision. Section 2.3 introduces the usage of Self Supervised Learning (SSL) in natural images and provides an overview of the relevant scientific literature related to SSL in RSIs, together with a review of the SOTA approaches in this domain. The following sections consider different Computer Vision (CV) tasks to approach the Weakly Supervised learning (WSL) problem using RSIs. Section 2.4 introduces the usage of WS for OD and provides an overview of the relevant scientific literature related to Weakly Supervised Object Detection (WSOD) in RSIs and the current SOTA approaches in this domain. Section 2.5 surveys Instance Segmentation (IS) and Weakly Supervised Instance Segmentation (WSIS). Finally, Section 2.6 surveys previous works on illegal landfills detection.

2.1. From Natural Images to Remote Sensing Images

According to Internet trend analysis, over the years the usage of Social Media and other Internet Services continuously increased, and the number of images available on the Internet increased accordingly. The massive number of images present on the Internet nowadays allows extracting some really valuable information that can be used in order to address several tasks [9, 23–26]. In particular, DL approaches rely on the availability of large amounts of images. At the same time, given the heterogeneous types of available images, it is fundamental to keep into consideration the characteristics of the used data when solving a specific problem.

2.1.1. Natural Images

The most common type of image is *natural images*. The availability of a large number of natural images has led to the possibility of extracting some precious information that can be used to address several tasks, such as image classification [9], object detection [23, 27], and instance segmentation [24]. Because of the impressive results obtained in these tasks, a lot of effort has been devoted to constructing huge natural images data sets with the aim of helping the development and comparison of novel solutions. These data sets are usually annotated at different granularities, to be used for several tasks, exploiting off-the-shelf softwares [28]. The most common annotation types for images are:

- **Image-level**: the label tells if an object of a certain class is present or not in the image. A slight variation of this annotation is the *scene-level* label, which records only the category of the most dominant object in the image.
- **Instance-level**: Polygons are drawn to delineate the boundaries of objects. Furthermore, every polygon is associated with a specific class. The most used shape is a simple rectangle (bounding box), but other shapes are used when it is necessary to take into consideration very specific object shapes or tasks [29].
- **Pixel-level**: in this case, the class is specified for every pixel, resulting in a fine-grained annotation of the image. Usually, a *background* class is introduced to account for pixels that do not belong to any of the other classes.

Similar to the image-level label, it is possible to use a finer-grained annotation named *region-level*, suggesting the presence or absence of at least one instance of an object in a portion (region) of the image. Other forms of labels exist, e.g., *point-based* annotations and *scribble-based* annotations. Another possible annotation type, that can be useful in case of the presence of multiple object instances in the image, is the *count* of the instances of each category inside the image.

Natural Images Data Sets

A lot of natural images data sets are presented in the literature. The most common ones, used as benchmarks for previously cited CV tasks, are reported in this section with an analysis of their peculiarities. It is important to notice that most of the reported data sets have multiple versions updated through the years, and for this reason, the reported statistics could depend on the version.

CIFAR-10 [30] is one of the most famous data sets used to evaluate classification methods. It consists of 60,000 32x32 color natural images divided into 10 classes, with

6,000 images per class. The following mutually exclusive classes are present: *airplane*, *automobile*, *bird*, *cat*, *deer*, *dog*, *frog*, *horse*, *ship*, *truck*. Every image is annotated at *image-level*.

ImageNet [31] provides an accessible image database that is organized according to the WordNet hierarchy. Each meaningful concept in WordNet, possibly described by multiple words or word phrases, is called a "synonym set" or "synset". There are more than 100,000 synsets in WordNet where ImageNet provides an average of 1,000 images to illustrate each synset in the WordNet. It offers tens of millions of cleanly sorted natural images for most of the concepts in the WordNet hierarchy. Every image is annotated at *image-level* and there are also *bounding boxes* for over 3,000 synsets. This makes this data set one of the most used ones for network pre-training, evaluation of classification, object localization, and object detection methods.

PASCAL VOC 2012 [32] is a data set containing natural images of vehicles, households, and animals in 20 object categories: *airplane*, *bicycle*, *boat*, *bus*, *car*, *motorbike*, *train*, *bottle*, *chair*, *dining table*, *potted plant*, *sofa*, *TV/monitor*, *bird*, *cat*, *cow*, *dog*, *horse*, *sheep*, *person*. Each image in the data set has *pixel-level*, *bounding box-level*, and *image-level* annotations. The PASCAL VOC 2012 data set extends the PASCAL VOC 2007 data set, resulting in a larger scale data set that consists of 11,540 images for training and 10,991 images for testing. It is one of the most used data sets to evaluate the performances of classification, object localization/detection, and semantic/instance segmentation methods.

MS-COCO [33] or simply COCO, is a large data set with more than 300,000 natural images (more than 200,000 of them are completely labeled). There are annotations at *image-level*, *bounding box-level* and *pixel-level*. In the 2014 version of the data set, there are more than 200,000 images covered by 80 object categories. It is one of the most used data sets to evaluate the performances of classification, object localization/detection, and semantic/instance segmentation methods.

2.1.2. Remote Sensing Images

Some research fields such as the medical field, require specific types of images that are more difficult to be retrieved, annotated and may introduce new challenges (e.g., class imbalance, label noise, heterogeneous organs, and lesions appearance [10, 25]) that can affect the performance of generic DL models. In this case, novel solutions need to be developed to account for the characteristics of this data.

This problem also arises when applications are developed in the RS domain, with images acquired by satellites or aerial devices.

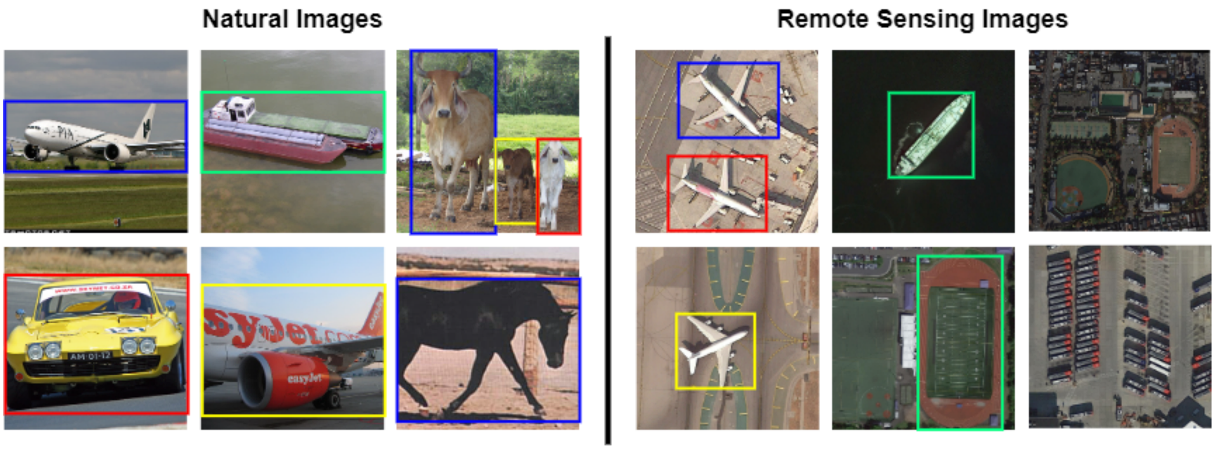


Figure 2.1: Example of natural images from PASCAL VOC data set [32] and RSIs from DIOR data set [34] and DOTA data set [35]. It is possible to observe that natural images usually contain few large objects while RSIs contain multi-scale, arbitrarily oriented, dense objects. Image taken from [13].

RSIs differ significantly from natural images, as shown in Figure 2.1 and Figure 2.2, for some aspects:

- Object instances only occupy a small portion of large images, while in natural images, few big objects are usually present.
- The background is complex and cluttered with the coexistence of multiple ground objects.
- Objects (e.g., ships and vehicles) can be extremely small and dense, while at the same time, large objects (e.g., ground track fields) can cover a vast area.
- Objects can have arbitrary orientations while they often appear with horizontal orientation in natural images.
- There is high intra-class diversity and inter-class similarity.
- RSIs generally capture the roof information of the geospatial objects, whereas natural images usually capture the profile information of the objects.

Therefore, as explained by Li et al. in [34] it is not surprising that the models learned from natural images are not easily transferable to RSIs. In the last few years, the increasing availability of RSIs allowed the application of DL methods to solve specific tasks such as airplane detection [37] and ship detection [38–40].

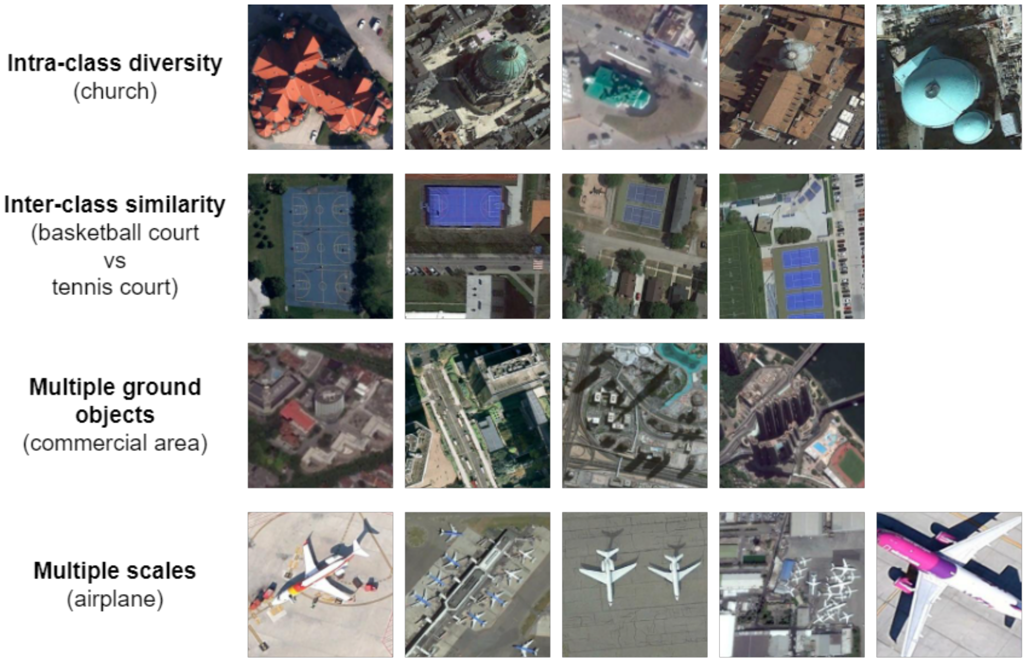


Figure 2.2: RSIs challenges on images from the NWPU-RESISC45 data set [36]. Image taken from [13].

Remote Sensing Images Data Sets

As for natural images, annotated data sets were also constructed for RSIs. Unfortunately, common RSIs data sets are not very widespread, and custom application-specific data sets are often built, thus limiting the possibility to compare techniques on common data. However, it is possible to recognize some data sets used in a decent amount of OD, SS, and IS approaches. Table 2.1 reports the most relevant statistics about the reported RSIs data sets. From a literature survey, *DIOR* and *NWPU-VHR-10.v2* are the most used to evaluate novel techniques on RSIs.

The data set named **Google Earth** is a collection of 120 high-resolution images of airports collected using the homonym service. It was proposed by Zhang et al. [37] to demonstrate that their algorithm can deal with multi-size targets in large-scale RSIs with cluttered backgrounds. Zhang et al. [42] extended the airplane detection task to include also vehicles and airports and incorporated images from **ISPRS** and **Landsat-7 ETM+**. The ISPRS data set provides vehicles with 100 very high-resolution images provided by the German Association of Photogrammetry and Remote Sensing [41]. The Landsat-7 ETM+ data set is acquired by the homonym sensor and includes 180 infrared RSIs of a variety of airports in China [42].

NWPU VHR-10 [43] is a ten-class geospatial object detection data set, containing

Table 2.1: Summary of the main data sets for RSIs.

Name	Year	Annotation Type	Number of Images	Number of annotations	Number of classes	Dimension (pixels)	Spatial Resolution	Target Area (pixels)
ISPRS [41]	2010	BB	100	-	1 (Vehicle)	≈ 900x700	8-15 cm	1150 ∼ 11976
Google Earth [37]	2013	BB	120	-	1 (Airplane)	≈ 1000x800	≈ 0.5 m	700 ∼ 25488
Landsat-7 ETM+ [42]	2014	BB	180	-	1 (Airport)	400x400	30 m	1760 ∼ 15570
NWPU-VHR-10 [43]	2014	Image, BB	800	3775	10	533×597 ∼ 1728×1028	0.08-2 m	1122 ∼ 174724
NWPU-VHR-10.v2 [44]	2017	Image, BB, Pixel	1172	-	10	400x400	-	-
DOTA [35]	2018	Image, Oriented BB	2806	188282	15	≈ 4000x4000	-	-
LEVIR [45]	2018	Image, BB	21952	11028	3	600x800	0.2-1 m	10 ∼600
TGRS-HRRSD [46]	2019	BB	26772	55740	13	-	0.6-1.2 m	-
WSADD [47]	2020	Image, BB	700	-	1 (Airplane)	768x768	0.3-2 m	-
DIOR [34]	2020	Image, BB	23463	192472	20	800x800	0.5-30 m	-

images from Google Earth and the German Association of Photogrammetry and Remote Sensing [41]. The classes are *Airplane*, *Ship*, *Storage Tank*, *Baseball Diamond*, *Tennis Court*, *Basketball Court*, *Ground Track Field*, *Harbor*, *Bridge* and *Vehicle*.

NWPU-VHR-10.v2 [44] is a ten-class geospatial object detection data set obtained by cropping images from the data set NWPU-VHR-10. In particular, 1,172 images of 400×400 pixels were obtained by cropping the positive images of the NWPU VHR-10 data set in which image sizes are different. The number of classes is left unmodified. The images are manually annotated at *image-level*, *bounding box-level*, and *pixel-level*.

DOTA [35] is a data set containing oriented BBs as annotations, presented as a large-scale benchmark data set and an OD challenge. Fifteen categories were annotated: *plane*, *ship*, *storage tank*, *baseball diamond*, *tennis court*, *swimming pool*, *ground track field*, *harbor*, *bridge*, *large vehicle*, *small vehicle*, *helicopter*, *roundabout*, *soccer ball field* and *basketball court*.

LEVIR [45] is a data set with a large number of high-resolution Google Earth images with over 22,000 images of 800×600 pixels and space resolution ranging from 0.2 m/pixel to 1.0 m/pixel. It can be used to evaluate OD approaches. LEVIR covers most types of ground features of the human living environment, e.g., city, country, mountain area, and ocean. There are 3 classes: *airplane*, *oil plot*, *ship*. The images are annotated at *image-level* and *BB-level*. It is important to understand that LEVIR is different from

LEVIR-CD [48], which is a remote sensing building Change Detection data set. They have a similar name because the authors share the same laboratory (Learning, Vision and Remote Sensing Laboratory).

TGRS-HRRSD [46] is a data set proposed analyzing the NWPU VHR-10 data set and recognizing that it is imbalanced and not large enough to train a CNN framework without strong augmentations. At the same time, it was identified that most of the existing algorithms are designed based on the assumption that the training data set is balanced, while the NWPU VHR-10 cannot quite fit this hypothesis. TGRS-HRRSD data set was proposed to try to address these problems.

WSADD is an airplane detection data set proposed by Wu et al. [47]. The images in this data set include airports and nearby areas of different countries (mainly from China, the United States, the United Kingdom, France, Japan, and Singapore) taken from the Google Earth satellite. It can be used to evaluate OD approaches, particularly aircraft detection approaches. Images are taken at different daytimes, seasons, and light intensities to ensure that the data set has a high diversity. The images are annotated at *image-level* with a label indicating if at least an airplane is present or not and at *BB-level* (in case of the presence of airplanes).

DIOR [34] is one of the largest, most diverse, and publicly available object detection data sets in the earth observation community that is often used as a benchmark to evaluate OD methods. It is a particularly challenging RSIs data set due to the variety of object sizes and different imaging conditions like weather conditions and seasons. The classes are: *airplane*, *airport*, *baseball field*, *basketball court*, *bridge*, *chimney*, *dam*, *expressway service area*, *expressway toll station*, *golf course*, *ground track field*, *harbor*, *overpass*, *ship*, *stadium*, *storage tank*, *tennis court*, *train station*, *vehicle*, *wind mill*. It is important to notice that DIOR has high inter-class similarity and intra-class diversity, and the number of object instances per class is not balanced. Every image in the data set is annotated at *image-level* and *BB-level*.

2.2. From Full Supervision to Weak and Self Supervision

ML is a sub-field of AI that develops solutions that do not rely on explicitly programmed instructions to perform a particular task but are able to improve by exploiting the information that can be extracted from training data. In particular, in the field of ML, DL methods that rely on deep neural networks to learn data representations, have gained much success in the last few years. The huge amount of available data (Section 2.1) and computational resources have led to a continuously increasing interest in extracting relevant and useful information from the data itself. According to the class of methods that is used, different types of information needs to be provided to solve ML tasks. There are several different classes of ML tasks:

- **(Fully) Supervised Learning**: it is a class of ML tasks that require annotated data to train ML architectures. This means that to address supervised tasks, it is necessary to provide not only the data itself but also the information concerning the target to be predicted. For instance, in the case of many CV tasks such as image classification, it is necessary to specify the label associated to each sample, together with the image itself. This class of approaches is further analyzed in Section 2.2.1.
- **Unsupervised Learning**: it is a class of ML tasks that allow learning underlying patterns or better representations of the data exploiting only the data itself without the need for any additional information.
- **Reinforcement Learning**: it is a class of ML tasks in which an agent tries to learn the optimal behavior that allows achieving a certain goal using feedbacks from its own actions and interactions with the environment in which the agent is placed.

Besides these three basic ML paradigms, it is also possible to identify some other categories of tasks that are much useful in special scenarios:

- **Weakly Supervised Learning (WSL)**: it is a class of ML tasks where limited supervision is used for labeling large amounts of training data in a supervised learning setting. It is used when noisy, limited, or imprecise labels are provided as supervision. This definition will be clarified in Section 2.2.2.
- **Semi-Supervised Learning**: it is a class of ML tasks that falls between Supervised Learning and Unsupervised Learning. In particular, it is concerned with the usage of a small amount of labeled data and a large amount of unlabeled data. For this reason, it is also a special instance of WSL. Semi-Supervised Learning aims to label unlabeled data by exploiting the knowledge learned from the few labeled data points.
- **Self-Supervised Learning (SSL)**: it is a class of ML tasks where no label is provided. However, differently from Unsupervised Learning, which tries to find high-level patterns, Self-Supervised Learning attempts to solve tasks that are traditionally targeted by Supervised Learning, without any labeling available. This class of approaches will be further analyzed in Section 2.2.3.

2.2.1. Fully Supervised Learning

Fully Supervised Learning methods [49–52] are based on the Supervised Learning framework. The main peculiarity of these approaches is that the whole data set needs to be finely annotated.

The level of detail of the labels depends on the specific task to be solved. For instance, to train a classifier [53–55], it is necessary to specify the class to which each training sample belongs (*image-level label*), whereas to train an object detector [49–51, 56, 57], it is necessary to specify also the location of each object, for instance, using *BB-level annotations*. However, annotating data is a time-consuming activity, especially in the case fine-grained annotations are required [11]. As an example, annotating images is quite fast if only image-level labels are required. On the other hand, if BB annotations are requested, the labeling process takes more time. If pixel-wise annotations are required, the process is even more expensive: labeling a BB on an object takes 10.2 seconds on average while labeling at pixel-level an object takes 79 seconds, which is about eight times slower [58]. For this reason, finely annotating whole data sets requires a huge effort. As a consequence, in some cases, only a small portion of the data set is annotated finely, whereas the remaining part is annotated using coarse-grained annotations. This is a major problem, especially for DL models that require lots of data to be trained. Given this fact, it is necessary to move to a new class of approaches that aim to perform the same supervised tasks, relying on approaches that do not require extensive supervision. This has led to advances in sub-fields such as transfer learning [59], semi-supervised learning [26], WSL [12, 13], unsupervised learning [60] and SSL [14, 15]. In this work, only WSL and SSL are analyzed in detail.

2.2.2. Weakly Supervised Learning

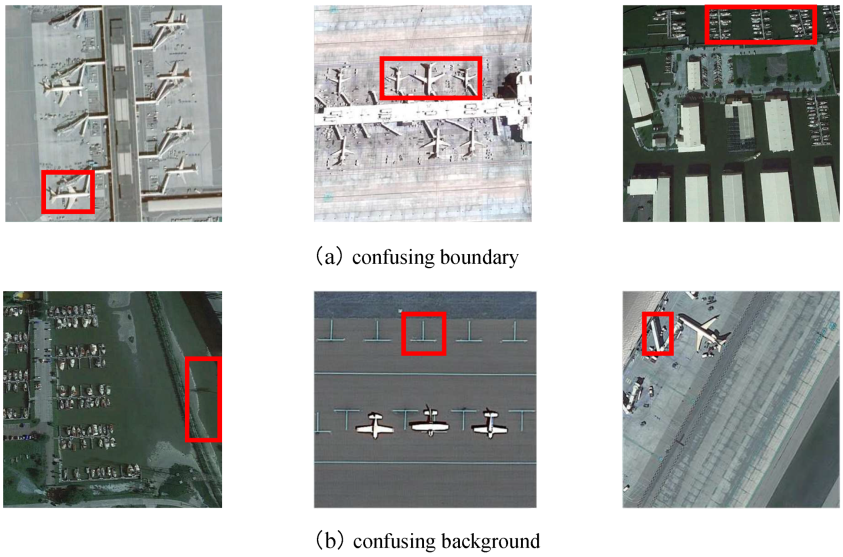
Weak Supervision (WS) is the branch of ML where limited supervision is used for labeling large amounts of training data in a supervised learning setting. Different types of weak supervision can be distinguished [61]:

- **Inexact supervision**: in this case, only coarse-grained labels are exploited as supervision. Using this type of weak supervision, it is possible to address the same supervised tasks using a weaker type of annotations. In particular, in the case in which input data are images, thanks to WS and the availability of huge data sets, classical CV tasks such as OD, SS, and IS can be accomplished using coarse-grained labels such as image-level ones [12, 13].

- **Inaccurate supervision**: in this case, labels used as supervision are noisy, partially unreliable, and sometimes not correct.
- **Incomplete supervision**: in this case, only a subset of training data is labeled. Approaches based on semi-supervised learning are usually considered part of this category.

In this work, only inexact supervision is taken into consideration. It is important to highlight the fact that the application of these methods is not restricted to images only and, as a consequence, not only to natural images. WS methods can also be applied to RSIs [47, 62–65]. For the RSIs domain, the usage of WS is even more important than it is for the natural images one: RSIs usually have a wide range of views, which means they contain much more objects of interest than natural images and this leads to higher labeling cost [66]. Nonetheless, this scenario is even more challenging due to the characteristics of RSIs (Section 2.1.2). As shown in Figure 2.3 it is harder to localize object instances because they often occur close to each other and only occupy a small proportion of large images with complex backgrounds. Moreover, under a low spatial resolution, it becomes difficult to identify smaller objects while under high spatial resolutions, objects can be misinterpreted as background. Thus, it is important to take into consideration these aspects when applying these methods since they have a huge impact on overall performance.

In the next sections, an analysis of the SOTA methods for the application of WSL to different CV tasks is performed. In particular, Section 2.4 introduces WS for the OD task while Section 2.5 does the same for IS.



(a) confusing boundary

(b) confusing background

Figure 2.3: Examples of localization issues introduced by RSIs. Image taken from [67].

2.2.3. Self-Supervised Learning

Self-Supervision is the branch of ML where a representation of the data is learned without the need for any type of human annotations. The idea behind this approach is that unlabeled data are exploited during training to extract as much information as possible to characterize the data itself. In particular, Self-Supervised Learning (SSL) can be considered a special type of unsupervised learning, given that no label is provided for the training set. However, as reported in [15] it is very ambiguous in various communities the difference between self-supervised and unsupervised learning. In this work, the same separation of the two terms adopted by Wang et al. [15] is used. More in-depth, traditional unsupervised approaches tend to utilize input data statistics and generate groups exploiting techniques such as dimensionality reduction [68, 69] and clustering [70]. On the other hand, SSL is a recent terminology that refers to a class of approaches in which a model is trained to learn good data representations either using supervision signals that are automatically generated from the data itself or maximizing the similarity between semantically identical inputs.

At the same time, SSL can also be considered a special type of semi-supervised learning in the sense that it is possible to fine-tune the model on a certain task using only a small amount of labeled data thus resulting in the usage of both labeled and unlabeled data as in semi-supervised learning. The SSL pipeline is similar to that used in the case of Transfer Learning (TL) (Figure 2.4) as reported in [14]. In particular, TL [59] is a solution for constructing data representations when the number of samples in the training set is limited. The basic idea is to transfer knowledge from a source task to a target one named *downstream task*. In particular, weights from a model trained on the source task are extracted and used as weights initialization for the target task model before eventually fine-tuning it. In traditional TL, the model is initially trained on a large labeled data set (e.g., ImageNet [31]), and then it is considered as a starting point to perform the target task training, without learning from scratch.

However, one of the major limitations of TL is that it usually works only if the source and target tasks are similar enough given that otherwise, the learned weights may not generalize well to the target task [14]. For instance, TL is less effective if the source and target tasks domains are much different. This can be the case for instance of a source task trained using natural images and a target task that is based on the usage of RSIs. At the same time, given that traditionally TL is performed using as a source task a supervised learning task (e.g., Image Classification) trained using a huge labeled data set (e.g., ImageNet [31]), it is difficult to exploit this solution when there is a significant lack of labeled data or when the labeling cost is high, such as in the case of RSIs. To solve these limitations, SSL can become a viable option to perform TL. In particular, while traditional TL still requires labeled data to extract the knowledge to be transferred, SSL only exploits unlabeled data to learn the representation. Furthermore, the data set that is used to learn the representation of the data is the same as that used to fine-tune the target task, solving the need of having source and target tasks that do not differ too much. Thus, it is possible to exploit both labeled and unlabeled data from the same data set, using the latter to solve a source SSL task, named *pretext task*, while the former is used to fine-tune the downstream task exploiting the knowledge gained from unlabeled data. SSL is a wide field of research both for natural and RS domains, and an in-depth analysis of the main techniques and the SOTA approaches will be performed in Section 2.3.

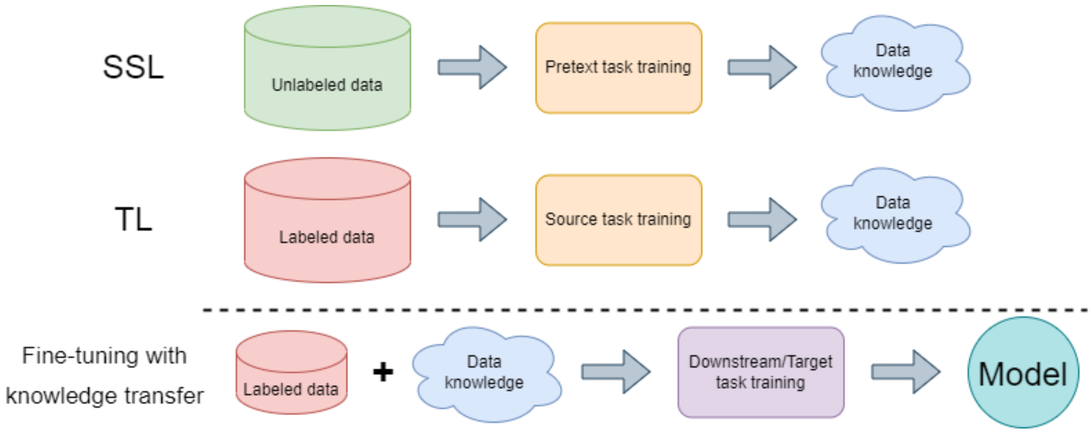


Figure 2.4: Visual comparison of SSL and TL. In the TL pipeline, a large amount of labeled data is used to train in a supervised manner a source task, while in the SSL pipeline a large amount of unlabeled data is used to train the pretext task using self supervision. In both cases, the extracted knowledge is combined with a small amount of labeled data to train a final model for the supervised downstream task. The more similar the data used in input for the source/pretext task is to the one used for the downstream task, the more effective the knowledge transfer will result.

2.3. Self-Supervised Learning

Self-Supervised Learning raised considerable attention in CV in the last few years and achieved significant milestones towards the reduction of human supervision [14, 15]. Indeed, it allows to extract representative features from unlabeled data, and outperform supervised pre-training on many tasks [71]. A good number of methods have been developed over the years to perform SSL. However, these methods can be categorized into three main classes of approaches:

- **Generative approaches**: in this case, representations are learned by reconstructing or generating input data.
- **Predictive approaches**: in this case, the representation is learned by predicting a self-produced label. This means that there is no need of annotating the data set since the label can be easily produced starting from the original input (e.g., when predicting the rotation angle of a rotated image, the rotation angle can be easily retrieved without the need for human annotations, starting from the unrotated image).
- **Contrastive approaches**: in this case, the representations of semantically similar inputs are compared and forced to be as much close to each other as possible.

Independently of the class of approaches that is used, once the representations are learned, the pre-trained model can be transferred to a downstream task. Eventually, fine-tuning can be performed as introduced in Section 2.2.3. As opposed to supervised pre-training, models pre-trained using SSL allow to leverage more general representations and ensure the possibility of collecting unlabeled data from the target task domain leading to a reduction of the domain gap between the pre-training phase and the downstream task training.

2.3.1. Generative Approaches

Generative approaches represent the first class of methods that have been used to learn better representations of the data. In particular, the basic idea is to learn the representation by performing input reconstruction as shown at the top of Figure 2.5 or generation. Two main approaches can be used to design generative solutions: Autoencoders (AEs) [72, 73] and Generative Adversarial Networks (GANs) [74]. AEs exploit an encoder-decoder architecture to reconstruct an input. The encoder network is used to generate feature representations (embeddings) that contain meaningful information about the input. Then, the decoder one aims at reconstructing the original input starting from the embedding. To avoid the trivial case in which the network learns the identity function, the embedding representation is much smaller than the input. The idea behind this kind of approach is that to be able to reconstruct the input, the network should have understood relevant aspects of the input and thus, the encoded features should be meaningful. Examples of these approaches include Variational AEs (VAEs) [75] and denoising AEs [72]. On the other hand, GANs use a generator network and a discriminator one trained in an adversarial way, to learn feature representations starting from input noises, which are essentially the feature representations of the output [14]. In particular, the input noise contains the information needed to generate the corresponding output. The vanilla GAN [74] was not designed for feature extraction. However, by inverting the generation process, it is possible to obtain a feature representation. Once the representation is learned, it is possible to keep only the network that produces the feature representation and use it as a feature extractor for the downstream task architecture as depicted in Figure 2.5.

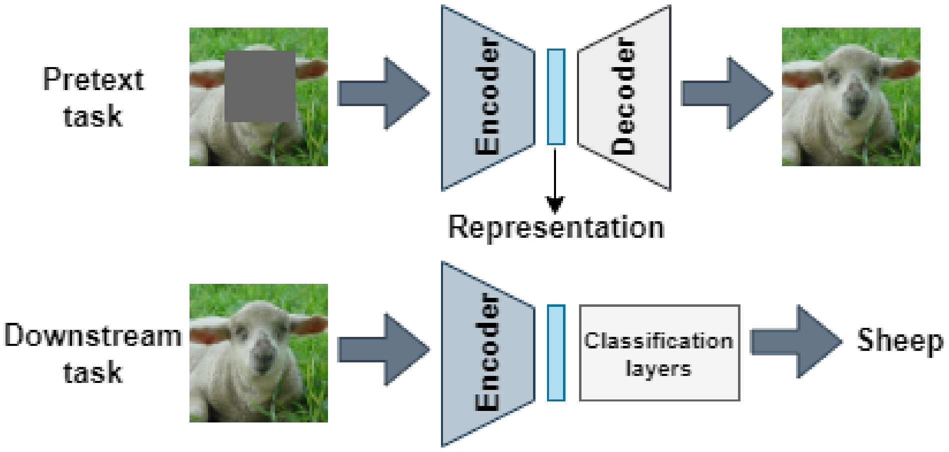


Figure 2.5: Example of SSL generative task based on image reconstruction. At the top, the pretext task is shown, while at the bottom, the encoder is kept to perform knowledge transfer to solve the downstream task of image classification. The used image is taken from Pascal VOC data set [32].

2.3.2. Predictive Approaches

Predictive approaches for SSL are based on auto-generated labels, that are used as a form of supervision to train the supervised pretext task. The idea is that a network could learn useful representations of the data while learning to predict specific properties of the data itself. Predictive SSL tries to solve the drawbacks of generative approaches [15]. More specifically, generative methods are based on pixel-level reconstruction. For this reason, pixel-level loss functions may overly focus on low-level details whereas in practice such details are not used by humans to recognize the contents of an image, since high-level details are usually more important. Moreover, pixel-based reconstruction does not typically consider long-range correlations that can instead be important for image understanding. Thus, providing the network with suitable high-level pretext tasks may allow high-level semantic information to be learned.

A predictive approach follows these steps:

1. Design a suitable pretext task for the data set.

2. Prepare self-generated labels.
3. Train the model to predict such labels and learn the data representation.
4. Use the feature extractor part of the network to transfer the knowledge to the downstream task.

One of the most intuitive examples of a pretext task designed for predictive SSL is predicting the rotation angle of an image [76] (see the top part of Figure 2.6). The idea is to rotate an input image by random multiples of 90° and assign a label representing the applied rotation. This label does not need to be annotated by humans given that it can be self-generated when the data set is loaded for training. The intuition is that to recognize the rotation, the network must be aware of the concepts of the objects depicted in the images. The same intuition holds also for other predictive pretext tasks. The main issue is that pretext tasks must be carefully designed taking into consideration both the domain and the downstream task, otherwise, pretext-specific representations could be learned, decreasing the network's generalizability. For instance, concerning the example reported at the top of Figure 2.6, it is easy to understand and predict the correct rotation angle by looking at the propeller which is an important part of the depicted object (see Figure 2.7). For this reason, predicting image rotations can be an adequate pretext task, that can potentially lead to the learning of significant data features like the propeller itself, that could be meaningful in a downstream task such as image classification. At the same time, predicting image rotations in the case of images such as the one reported in the bottom

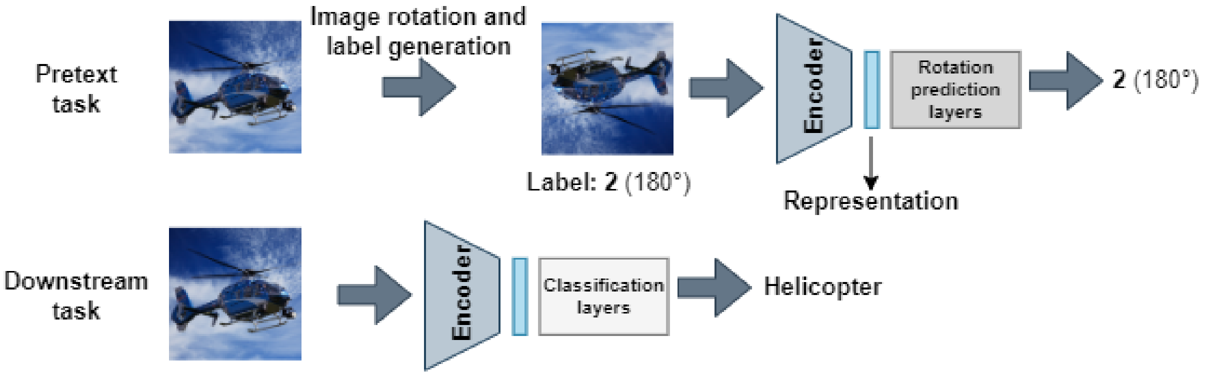


Figure 2.6: Rotation prediction task for an image of a helicopter. At the top, given the original image, a rotation is performed (180° in this case) and a label is assigned according to the applied rotation angle. The network is fed with the rotated image and with the corresponding label as supervision. In this example, since the image is rotated by 180°, the network should predict 2. At the bottom, the encoder is kept to perform knowledge transfer to solve the downstream task of image classification.

part of Figure 2.7 is much more difficult. Thus, predicting rotations is not a good choice in this second case. This could potentially lead to the failure of the SSL task. Given the sensitivity and importance of the pretext task design, various context information of the input data can be taken into consideration to design a pretext task suitable for the specific application. In this sense, pretext tasks can be categorized depending on the context information they exploit [15]:

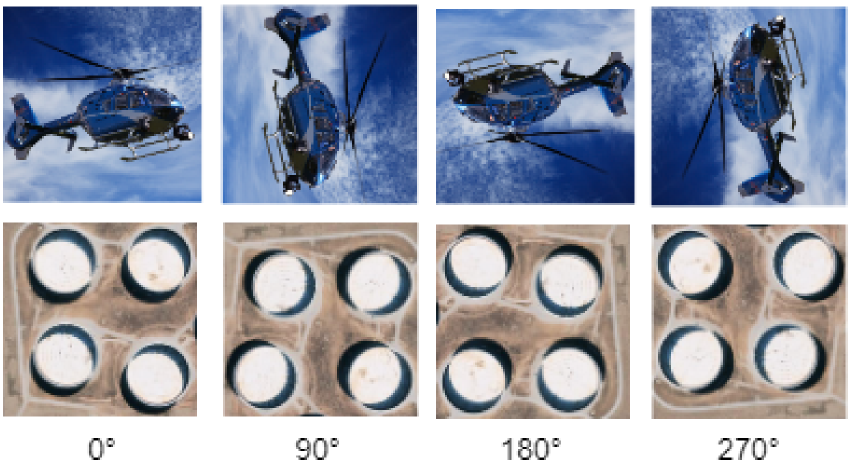


Figure 2.7: Image rotations for two different images. In the first case, it is quite easy to understand the degree of rotation of the helicopter, for instance by looking at the propeller. Instead, for the second image, understanding the rotation degree is much more difficult given that there are only minor changes in the rotated versions.

- **Spatial context**: the spatial information contained in the images is used to generate labels. An example, is the prediction of the image rotation [76].
- **Spectral context**: the spectral information contained in the image channels is used to generate labels. For example, the prediction of one spectral channel taking the others as input could be used as a pretext task [77].
- **Temporal context**: this is mainly related to the video domain, where relationships between different frames could be used to design specific pretext tasks [78].

Apart from these three families, other types of contextual information could be exploited. When designing the pretext task, additional care must be taken to ensure that the pretext task is solved by learning important features and not by finding a trivial solution. Otherwise, the gained knowledge is not useful for the downstream task. For example, in the case of [79], where an image is split into 9 tiles and the network has to predict the relative position of two tiles, the authors show that the network can find different trivial solutions. For instance, it is shown that the network could exploit low-level cues like boundary patterns or textures continuing between patches, to solve the task. However, this leads to learning something that is not useful to solve the downstream task, because this type of knowledge is not related to the data representation. Thus, ad hoc solutions should be explored to avoid this kind of situation.

Once the training of the pretext task is completed, the gained knowledge (representation) can be transferred to the downstream task using the same encoder of the pretext task as a feature extractor for the target task. If needed, fine-tuning can also be performed on the feature extractor.

2.3.3. Contrastive Approaches

Despite the good performances reached by predictive SSL approaches, the difficulty of designing suitable pretext tasks, taking into account the need to avoid possible trivial solutions and the effectiveness of the learned data representation, still represents an open issue. For this reason, the idea behind contrastive learning is to give the network more freedom to learn high-level representations, without relying on a single pretext task. More specifically, contrastive approaches train a model by contrasting two semantically identical inputs and pushing them to be close to each other in the representation space. In the case of images, semantically identical inputs can be computed by applying different combinations of augmentations to the input image (e.g., random crop, flip, color jitter, gaussian blur).

As shown at the top of Figure 2.8, the contrastive learning methods are usually built as a Siamese-like architecture [80]. However, one of the major problems of these approaches is *model collapse*, which is concerned with the fact that only enforcing similarity between pairs of input could lead to a trivial solution in which the model maps all of its input data to the same representation [81]. Depending on how the model collapse problem is handled, contrastive learning approaches can be categorized according to the following taxonomy: *negative samples*, *clustering*, *knowledge distillation* and *redundancy reduction*. The first solution proposed to the collapsing problem is the one based on the usage of negative samples [82–84], which consists in introducing dissimilar samples to have both positive and negative pairs. While the positive samples are different augmentations of the same image, the negative samples come from other data points in the data set and their selection is a critical process [85]. The presence of negative samples allows repulsing negatives while continuing to attract positive samples in the embedding space. Given an anchor data point x, a positive example x^+, a negative one x^- and an encoder E, the objective could be formalized as:

$$\operatorname{sim}\left(E(x), E\left(x^{+}\right)\right) \gg \operatorname{sim}\left(E(x), E\left(x^{-}\right)\right) \tag{2.1}$$

To achieve this for every data point x, a lot of approaches have been developed (see Section 2.3.4 and Section 2.3.5), and the focus of them is based on designing accurate loss functions, named *contrastive losses*, able to capture the desired objective. In this scenario, the contrastive learning framework could be summarized by the following pipeline:

1. **Data augmentation**: images are augmented in different ways changing the visual appearance but not the semantic content.
2. **Encoder**: feature representations are extracted from the augmented images exploiting a network named encoder.
3. **Projection Head**: the representations obtained through the encoder are mapped to the representation that is then used by the contrastive loss function.
4. **Contrastive Loss**: the loss function minimizes the latent embedding distance between positive pairs while simultaneously maximizing the distance between negative pairs.

Another sub-class of approaches [86, 87] is concerned with learning data representations by using a clustering algorithm to group similar features together in the embedding space. Given the absence of label supervision, the clustering process is used as a self-labeling mechanism to determine which samples should have a similar representation and which others shouldn't.

Knowledge distillation methods [88], instead, commonly make use of a teacher-student network [89] (that is still Siamese-like) and optimize a similarity metric of two augmented views of the same input image. Negative samples are not needed in this case because the knowledge transfer between the student network and the teacher one is based on asymmetric learning rules or asymmetric architectures.

Finally, the idea of redundancy reduction methods [90] comes from neuroscience and the intuition of Barlow et al. [91], which states that the goal of sensory processing is to record highly redundant sensory inputs into a factorial code. Thus, the principle of redundancy reduction was extended and used to avoid trivial solutions in contrastive learning, without the usage of negative samples.

In the contrastive framework, once the training procedure is completed, the knowledge is transferred to the downstream task using the same encoder of the contrastive learning framework as a feature extractor for the target task, as shown in Figure 2.8. It is also possible to perform fine-tuning on the feature extractor if needed.

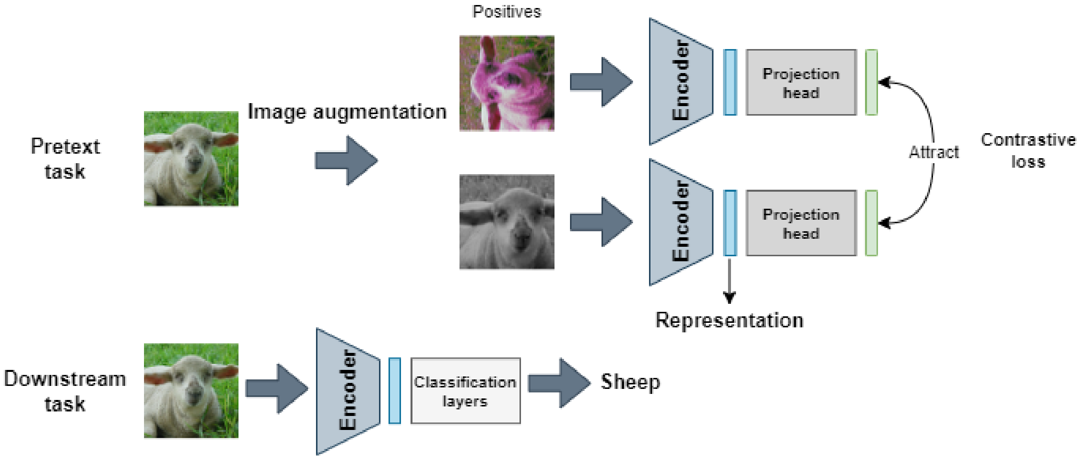


Figure 2.8: Contrastive learning approach for an image of a sheep. At the top of the image, two different data augmentations are initially performed on the same image obtaining semantically identical views. Following the Siamese-like architecture [80], images are projected into the embedding space, and, after the projection head, the contrastive loss is used to minimize the latent embedding distance between the representations of the two augmented views. The contrastive loss generally refers to the approaches that use negative samples. However, the pipeline is almost identical also for the other types of approaches, considering more suitable objective functions (e.g., similarity metrics for knowledge distillation). At the bottom, the encoder is kept to perform knowledge transfer to solve the downstream task of image classification. The used image is taken from Pascal VOC data set [32].

As stated by Wang et al. [15], in some literature, the term contrastive is only used to denote methods that include negative samples. However, in this work also methods without negative samples are considered part of this class of approaches. Figure 2.8 could be easily extended to the other presented types of contrastive learning since the overall pipeline is not much different than the depicted one and what changes is usually the objective function.

2.3.4. SSL in Natural Images

As already outlined in the previous sections, several different approaches for SSL have been developed over the years. In particular, generative methods learn to reconstruct or generate input data, predictive approaches learn to predict self-generated labels, and contrastive methods aim to maximize the similarity between semantically identical inputs. Generative approaches are the least recent approaches in the field of SSL. Many different architectures have been proposed during the years for both AEs [92] and GANs [74, 93].

In 2008, Vincent et al. [72] propose the use of Denoising AEs to learn useful feature representations. In this case, the input that is fed to the encoder is disrupted with some noise and the AE is expected to reconstruct a clean version of the input.

In 2011, Ng et al. [94] introduce the concept of Sparse AEs in which the embedding representation's size is bigger than that of the input instead of being smaller as it is most of the time. However, if the representation size is equal to or bigger than that of the input, the architecture could learn the trivial identity mapping. To avoid this situation, the authors propose to enforce the encoded representation to be sparse. During the same year, Contractive AEs are proposed by Salah et al. [95]. The emphasis is on making the feature extraction less sensitive to small perturbations of the input. This is done by forcing the encoder to disregard changes in the input that are not important for the reconstruction, by adding a penalty term to the loss function of the AE. The main difference between Contractive AE and Denoising AE [72] is that the latter encourages the robustness of the reconstruction, which can only partially increase the robustness of the representation, while the former encourages directly the robustness of the representation.

In 2013, Kingma et al. [75] introduce the concept of Variational AE which is a generative model that attempts to describe data generation through a probabilistic distribution. The idea is that instead of a single latent representation associated with each sample as was the case of previous methods, the encoder maps the input to a Gaussian distribution described by mean and standard deviation. Then, the decoder randomly samples a latent vector from the distribution to reconstruct the input.

In 2014, Goodfellow et al. [74] introduce GAN. The basic architecture of a GAN is composed of a generator network and a discriminator one. The former aims at generating new samples (fake samples) similar to those of a specified data set (real samples), starting from noise, while the latter tries to distinguish between real samples and fake ones. The idea is that once the generator can fool the discriminator, it is possible to discard the discriminator and use the generator to produce new samples. By inverting the generator, it is instead possible to obtain a representation vector of a sample. For this reason, in 2016, Donahue et al. [93] proposed Bidirectional GAN (BiGAN). In this case, besides the classic generator, another generator network is designed to move from the data to a latent vector. The discriminator in BiGAN discriminates between the couple input data and generated latent representation versus the generated data and input representation. During the same year, Pathak et al. [96] presented context encoders to generate missing regions within an image. To accomplish this task, the image with missing regions is fed to the context encoder, which outputs the missing pixels. The architecture is the typical encoder-decoder one. The idea is that the context encoder can predict the missing pixels if the content of the image is understood and thus a good representation is learned.

In 2017, Zhang et al. [97] propose Split-brain AEs designed for the task of representation learning. The idea is to split the typical AE network into two disjoint sub-networks, where each sub-network is trained to perform a difficult task named cross-channel prediction. This complex task is the prediction of one subset of the data channels from another subset. Together, the two sub-networks are able to extract features from the entire input. The authors show that by forcing the network to solve the cross-channel prediction task, the learned representation transfers well to other unseen tasks.

More recently, He et al. [73] propose Masked AEs (MAEs) to perform self-supervised representation learning. Inspired by Denoising AE [72], MAE masks out random patches of the input image, feeds visible patches to the encoder, and reconstructs the missing patches starting from the latent representation and masked tokens. This work is based on transformers [98], thus proving their potential for self-supervised visual representation learning.

While being effective, generative approaches perform pixel-level reconstruction or generation. However, being able to generate very high dimensional data points (e.g., whole images) is not necessary to learn a good representation for many downstream tasks. In fact, instead of focusing on the whole sample, it is possible to learn useful representations by focusing only on the prediction of specific properties of the data. For this reason, around 2015, several researchers started to design suitable pretext tasks for representation learning.

Doersch et al. [79] propose to decompose the input image in 9 tiles and predict the relative position of the middle tile with respect to another one. In the paper, the authors focus also on tackling the problem that the network can find trivial solutions (e.g., using chromatic aberration) by adding a gap between patches and dropping color channels. Noroozi et al. [99] propose another pretext task following a similar idea. In particular, instead of predicting the relative position of two tiles, this method focuses on solving jigsaw puzzles with 3 × 3 patches. Thus, a permutation of the original image patches is given as input to the network, whose aim is to predict which permutation was performed. Variations of this pretext task exist [100–102]. The same authors also propose another pretext task based on counting the objects in the image [103].

Zhang et al. [77] propose to let a network learn to colorize an image starting from an uncolored one. This has been proven to be effective, especially in the case of semantic segmentation downstream tasks [104].

In 2018, Gidaris et al. [76] propose to predict the rotation angle of an image to learn good representations of the data. The authors found that the training was significantly improved by feeding four rotated images into the network simultaneously, instead of a single, randomly rotated image. Other pretext tasks have been proposed over the years. For instance, in 2019, Kim et al. [105] proposed to predict the order of the frames of a video. This type of pretext task thus considers temporal context.

The performance of predictive SSL depends largely on a good pretext task. However, as already pointed out in the previous subsection, given that designing suitable pretext tasks is complex and may even lead to pretext-specific representations, contrastive methods have been proposed. Concerning contrastive methods, most of the approaches are based on instance discrimination, an approach that classifies each image separately and aims at finding suitable features representation by looking at the single instances. These methods initially make use of negative sampling to avoid the problem of model collapse. Instance discrimination was explored by Wu et al. [106]. The idea behind this work is to use a network to encode each image as a feature vector, which is projected to a 128-dimensional space. Then, the optimal feature embedding is learned via instance-level discrimination, trying to maximally scatter the features of training samples in the 128-dimensional space. The features are memorized inside a *memory bank*.

Misra et al. [82] propose a Pretext-Invariant Representation Learning (*PIRL*) method based on the observation that most previous predictive methods [76, 79, 99] learn representations that are somehow dependent on the specific transformation that is applied, and not invariant. Thus, PIRL feeds a Siamese network with an image and one augmented view of the same image and forces their representations to be similar. At the same time, a memory bank is kept following the idea of [106] to memorize the feature representations of negative samples to be used for contrastive learning. The memory bank contains a moving average of representations for all (non-augmented) images in the data set. Usually, if memory banks are not used, the negative samples that are used are those contained in the same mini-batch. However, given that contrastive methods tend to work better with a large number of negative samples, the batch size should be quite large. Thus, from this point of view, memory banks can be more efficient. However, at the same time, maintaining a memory bank during training can be a complicated task, as updating the representations in the memory bank can be computationally expensive since representations get outdated quickly, and the representations in the memory bank are always one step behind the current encoding generated by the network, leading to unwanted mismatches.

To address these issues, in 2020, He et al. [83] replaced the memory bank by a separate module called *momentum encoder*, proposing a method called Momentum Contrast (*MoCo*). The idea is to view contrastive learning as a dictionary look-up, building a dynamic dictionary with a queue containing the encoded features of previous mini-batches that are updated using a moving-averaged encoder. Then, the current mini-batch is enqueued to the dictionary while the oldest one in the queue is removed. In this way, the issues concerned with memory banks can be solved, resulting in a performance improvement.

Based on previous works, Chen et al. [84] proposed a milestone of contrastive learning called *SimCLR* (SIMple framework for Contrastive Learning of visual Representations). SimCLR makes use of a classical end-to-end architecture that is fed with two augmented views of the same image. Moreover, large batch sizes are used to handle the performance bottleneck related to the number of negative samples. However, the main contribution of SimCLR is that it illustrates the importance of the used data augmentations. Furthermore, SimCLR also includes an additional learnable non-linear transformation (projection head) between the representation that is then used in the downstream task and the one used in the contrastive loss. The authors highlight the fact that using the representation before the projection head is more effective for the downstream task since the second representation loses information because of the contrastive loss. More specifically, the representation after the projection head is trained to be invariant to data transformation. However, this may remove information that can be useful for the downstream task, such as the color or the orientation of objects.

Another set of contrastive learning methods aims to learn data representation by using a clustering algorithm to group similar features together in the embedding space. Following this idea, Caron et al. [86] propose a method called *DeepCluster* based on leveraging K-means clustering [107] to generate pseudo labels. Then, these assignments are used as supervision to update the weights of a network. Finally, this process (clustering and network training) is iterated. However, one of the major limitations of this approach is that the two-stage training is time-consuming and less effective compared to instance discrimination-based methods such as SimCLR [84] and MoCo [83], which do not use any clustering stage and make use of data augmentations to boost the performance. At the same time, these methods require explicitly computing a large number of pairwise feature comparisons.

For these reasons, in 2020, the same authors decide to make use of online clustering and multi-view data augmentation, proposing a new method called *SwAV* (SWapping Assignments between multiple Views). SwAV exploits the advantages of instance discrimination-based methods without requiring to compute pairwise comparisons. The method simultaneously clusters the data while enforcing consistency between cluster assignments produced for different augmentations of the same image, instead of comparing features directly. The intuition is that, given some clustering centroids, different views of the same images should be assigned to the same centroids. This method is more efficient than the previous ones given that it requires neither a large memory bank nor a special momentum network. Based on SwAV, Goyal et al. [71] demonstrated the effectiveness of SSL in real-world scenarios by surpassing for the first time the best supervised pre-trained model.

Another set of methods for contrastive SSL is based on knowledge distillation [108]. In 2020, Grill et al. [88] proposed *BYOL* (Bootstrap Your Own Latent). The general architecture is similar to that of MoCo [83] in which however no negative samples are used. BYOL relies on two neural networks, called online (teacher) and target (student) networks respectively, that can interact and learn from each other. The idea is that starting from an augmented view of an image, the online network is trained to predict the representation of the same image under a different augmented view produced by the target network. The target network is updated in a similar way to MoCo, using a slow-moving average of the online network. To prevent the problem of model collapse, BYOL makes the two networks asymmetric by introducing an additional predictor on top of the online network. This method represents a milestone for further approaches [109]. Furthermore, recent works [110, 111] have improved the performances by exploiting also image transformers [98].

Finally, the last sub-class of contrastive methods is based on the idea of redundancy reduction [91]. In particular, Zbontar et al. [90] proposed Barlow Twins. In this case, redundancy reduction is used as a way to avoid model collapse without the need of using negative samples. The overall architecture is similar to the typical contrastive learning network (see the top of Figure 2.8). However, the objective function measures the cross-correlation matrix between the embeddings of two identical networks fed with augmented views of a batch of samples and tries to make this matrix close to the identity. In this way, the representations of semantically identical inputs are forced to be similar, while, at the same time, minimizing the redundancy between the components of these vectors. This method as well as others such as BYOL [88] are also more robust to the choice of the batch size than other methods such as SimCLR [84], in which a big batch size is needed to obtain better results. At the same time, these methods do not even need to make use of additional memory banks such as in [82, 106] given the fact that negative samples are not used. This allows further boosting of the performances of contrastive SSL approaches.

2.3.5. SSL in Remote Sensing Images

The approaches described in Section 2.3.4 have been trained on natural images. However, the application of SSL methods is not limited to this type of image, and it possesses a high potential in other more specific domains such as RS, especially because of the poor performances obtained with TL using natural images data sets (e.g., ImageNet [31]). The peculiarities of RSIs, described in Section 2.1.2, must be taken into account also in the case of SSL. For this reason, the design of a predictive pretext task can be even more difficult in the case of RSIs. For instance, as depicted in Figure 2.7, learning feature representations by predicting image rotations [76], may not always be a good choice. Another important consideration is that RSIs usually come with a larger number of spectral channels. Thus, barely using the raw natural images methods described in the previous subsection, which make use of RGB images, would lead to discarding all the information that can be derived from other channels [15]. As in the case of natural images, however, these methods can be grouped into the same three main categories as before: *generative approaches*, *predictive approaches*, and *contrastive approaches.*

Regarding generative approaches, AEs have been widely used to learn representations from RSIs [112–114]. More specifically, studying the change detection problem for images with various spatial resolutions, in 2016, Zhang et al. [112] propose a stacked denoising AE to learn image features. One year later, Lu et al. [113] exploit a shallow weighted deconvolution network to learn a set of feature maps and filters for each image by minimizing the reconstruction error between the input image and the convolution result. Then, a Spatial Pyramid Model (SPM) [115] is used to aggregate the obtained features at different scales and finally feed them to an SVM [54] for classification. An important application of AEs in RSIs is Hyperspectral Image (*HSI*) analysis, in which AEs are either used for pre-training [114], or exploited in downstream tasks.

Few GAN-based methods have been developed for SSL. However, several works try to integrate GANs to target applications [116–122]. For instance, Zhu et al. [116] exploit GANs to perform HSI classification, while Hughes et al. [118] propose a GAN-based framework to generate similar samples to a given image.

Regarding predictive SSL, several methods tend to exploit the spatial context of the

RSIs [123–127], given that this information is quite relevant. Among the approaches presented for natural images, jigsaw puzzles [99] are not widespread in SSL for RSIs given that the spatial correlation in aerial images is less dominant. In fact, in RSIs, translation invariance is prominent given that close patches tend to be similar to each other in several scenarios (e.g., water surfaces, deserts, forests, mountains, etc.), as reported in [15].

For this reason, specific predictive pretext tasks can be designed for RSIs. In 2018, Singh et al. [123] propose a pretext task for SS of aerial images based on image semantic inpainting, given the similarity between inpainting and the SS task. The SSL learning follows an adversarial training scheme, with a gradually increasing difficulty of the pretext task, leading to a better data representation.

During the next year, Zhang et al. [124] develop a Rotation Awareness-based learning framework termed *RotANet* for the task of automatic target recognition with Synthetic Aperture Radar (SAR) images. RotANet uses SSL to learn to predict a set of rotation angles given a sequence of rotated SAR-probed targets and autonomously generalize this ability to other sequences without external supervision.

In 2020, Tao et al. [125] design a pretext task for RSIs scene classification, based on both relative position [79] and image inpainting [96], with the addition of instance discrimination [106]. During the same, year Zhao et al. [126] adopt a mixup strategy for RSIs scene classification. More specifically, the authors design a multi-task framework able to combine the SSL pretext task, based on rotation [76], and the downstream task of RSIs scene classification, using dynamic weights. More recently, Ji et al. [127] approach the few-shot scene classification problem by exploiting both the rotation prediction pretext task and contrastive prediction pretext task during training.

Besides spatial context, in the case of RSIs, spectral context can also represent a valuable asset for the design of pretext tasks, because multiple spectral bands beyond the RGB color space are usually present. At the same time, exploiting these spectral bands for SSL can be challenging, and requires attention.

In 2020, Vincenzi et al. [128] propose a method that tries to learn meaningful representations from satellite imagery, leveraging the high-dimensionality spectral bands to reconstruct the visible colors. Moreover, the authors observe that predictions based on natural images and colorization usually rely on different parts of the input, thus using an ensemble model can improve the overall performance. One year later Wu et al. [129] propose the usage of SSL to train a deep network for hyperspectral dimensionality reduction.

Given that in RSIs video recording is not yet common as outlined in [15], the temporal context is still not much explored to design pretext tasks. However, RSIs taken at different times are very important for applications such as change detection [112]. Hence, in 2020, Dong et al. [130] propose an SSL technique for RSIs change detection using a GAN discriminator. During the same year, Yuan et al. [131] propose a transformer-based SSL method for the task of satellite time series classification. This work was recently further improved [132].

In natural images, contrastive learning is currently the class of approaches that provides the best overall performance for SSL. Thus, it is not surprising that the usage of contrastive learning for SSL in RSIs is one of the most promising branches of research and, as in the case of RSWSOD (see Section 2.4.6), methods developed for natural images such as MoCo [83] and SimCLR [84] could be a good starting point for further refinement and adaptation to the RS domain. The first SSL approach for RSIs that exploits a contrastive learning framework is *Tile2Vec*, proposed by Jean et al. [133] in 2019. This method extends the distributional hypothesis from natural language that words appearing in similar contexts tend to have similar meanings (that inspired Word2Vec [134]), to spatially distributed data. The basic idea is to exploit a triplet loss to move closer neighboring tiles (positive tiles) and move further the distant tiles (negative tiles) in the feature space. Moreover, the authors show that vector operations can be performed on the obtained representations as in the case of the representations produced by Word2Vec.

Many approaches have been developed based on the contrastive learning framework making use of negative sampling. Jung et al. [135] reformulate the triplet loss to binary classification loss, adding also no-updated fully connected layers to improve robustness, while Leenstra et al. [136] propose a combination of triplet loss and binary cross-entropy loss for SSL in RS change detection.

Other approaches follow a design similar to SimCLR [84] and apply it to perform HSI classification [137–139]. In particular, Zhu et al. [138] develop *SC-EADNet*, a network using a multi-scale feature extraction approach, while Zhao et al. [139] show promising results with very limited labels.

As analyzed by Wang et al. [15] all these HS data-related works rely on simple spatial and spectral augmentations (e.g., random cropping, Gaussian noise, etc.) to generate views of the same image. However, given the impact that data augmentation techniques can have on the final performance [84], there is still much room for improvement.

In 2020, Kang et al. [140] propose *SauMoCo*, exploiting the basic idea of Tile2Vec [133] and slightly modifying it to allow considering not only semantic similarities among nearby RS scenes but also the inherent semantic diversity of land cover concepts. In this case, MoCo [83] is used as a contrastive learning method.

In 2021, Jung et al. [141] combine the sampling idea of Tile2vec [133] with SimCLR [84], proposing *SimCLR with smoothed view*. The algorithm is based on spatial augmentation, and it simultaneously utilizes several neighboring images as a positive pair of the anchor image. Furthermore, the proposed approach uses multiple-input images and averages their representations (smoothed representation). More specifically, K-neighboring image representations, corresponding to positive images, are averaged to create a smoothed representation, which is useful for reducing the impact of noise.

During the same year, Li et al. [142] added a contrastive loss term between patches of an image in spite of using only a contrastive term between views of the same image as done by most of the approaches.

In 2022, Montanaro et al. [143] tackle the land cover classification task using SimCLR [84] for the representation learning of the encoder and a perturbation invariant AE for the segmentation training of the decoder. Scheibenreif et al. [144] address the same task using Swin Transformers [145] with a contrastive data fusion SSL strategy [146], showing that latent representations derived through SSL pre-training and subsequent supervised fine-tuning are task agnostic and can be utilized for both land cover classification and segmentation. Stevenson et al. [147] propose to use SimCLR [84] for representation learning of LiDAR elevation data, facing the complexity of this type of multi-dimensional data.

Apart from all of these methods [140–142, 144, 147], that consider the spatial contexts of RSIs to generate positive and negative pairs, Contrastive Multi-view Coding can be used for multispectral and HS representation learning [148–150].

In 2020, Ayush et al. [151] propose to exploit the spatiotemporal structure of RS data by combining contrastive learning and predictive learning. The authors leverage spatially aligned images over time to construct temporal positive pairs in contrastive learning and geo-location to design predictive pretext tasks. One year later, Heidler et al. [152] proposed an extension of the triplet loss to exploit the correspondence between geo-tagged audio recordings and RSIs. This loss is called *batch triplet loss* and it could be used for audio-visual multi-modal SSL.

Referring back to the contrastive SSL methods that approach the model collapse problem through clustering, several approaches have been designed also for the RS domain. In 2020, Walter et al. [120] presents a content-based image retrieval framework for RSIs, investigating the usage of SSL techniques such as DeepCluster [86], VAE [75], colorization as pretext task and BiGAN [93]l.

In 2021, Saha et al. [153] face the multi-sensor RSIs change detection task with a combination of images acquired by optical and SAR sensors using SSL. The authors propose to combine DeepCluster [86] and triplet contrastive learning. The authors further integrate DeepCluster [86], BYOL [88] and MoCov2 [154] in a derivative work [155]. More recently, Liu et al. [156] contribute to the design of a novel clustering-based contrastive loss to capture the structures of views and scenes, proposing a Dual Dynamic Graph Convolutional Network named *DDGCN*.

Considering knowledge distillation SSL methods, most of the designed approaches for RSIs [157–160] are based on BYOL [88] and its extensions such as SimSiam [109]. In 2021, Guo et al. [157] propose to combine GAN [74] and BYOL [88] for better discriminative representation learning. The authors add a similarity loss to the discriminator loss by seeing the discriminator as a self-supervised encoder, which encodes both fake and real images as two input views to a BYOL-like Siamese network. Following a similar intuition, Hu et al. [158] utilize a transformer as an encoder backbone combined with a BYOL [88] baseline structure to address the HSI classification task. More recently, Zhang et al. [161] also exploits an attention-based vision transformer, where global and local augmented views are contrasted based on self-distillation [162].

Muhtar et al. [159] propose *IndexNet*, an SSL method for SS with RSIs. IndexNet is built on BYOL [88] and performs contrastive learning at image and pixel levels to preserve spatial information. By combining image-level contrast and pixel-level contrast, IndexNet can learn spatiotemporal invariant features. Similarly, Chen et al. [160] propose a pixel-level SSL approach for change detection. This method is based on SimSiam [109] with the objective of enforcing point-level consistency across views. The authors also propose to use background-swap augmentation to focus more on the foreground.

Finally, regarding redundancy reduction approaches, methods for natural images such as Barlow Twins [90] are relatively new, and thus, up to now, these methods are directly applied also to RSIs [163, 164].

Thus, in general, RSIs SSL approaches are usually based on those developed for natural images, with the introduction of modifications to address RSIs challenges.

2.4. Weakly Supervised Object Detection

In the field of CV, one of the most studied tasks is that of Object Detection [23, 27]. OD is a CV technique that allows to identify (classify) and precisely locate objects in an image.

Usually, the location is delineated using BBs. Several methods have been developed in the last few years to address problems such as vehicle detection [165–167], airplane detection [37, 47] and ship detection [38–40].

In Fully Supervised Object Detection (FSOD), it would be necessary to annotate the data set specifying both the bounding box enclosing an object and the category to which the object belongs. Recently, successful FS approaches have been developed [49–51, 56, 57] to solve the task of OD. In particular, they can be categorized into two macro-groups:

- **Region proposal-based**: these methods use a two-step approach. First, they identify regions, where objects are expected to be found using off-the-shelf techniques, [168, 169] as done in [49, 50] or using a built-in Region Proposal Network (RPN) as in [51], to extract a set of candidate BBs that are highly likely to contain an object instance. Then, they detect objects only in the identified regions using a Convolutional Neural Network (CNN). Examples of these approaches are R-CNN [49], Fast R-CNN [50] and Faster R-CNN [51]. These methods are usually slightly better in terms of accuracy but slower due to the presence of the region proposal step.
- **Region free**: these methods recast the OD problem as a single regression problem, straight from image pixels to bounding box coordinates and class probabilities. They propose a fully convolutional approach in which the network is capable of identifying all the objects in a single forward pass over the image. These methods are thus usually faster than those based on region proposals. Examples of these approaches are YOLO [57] and SSD [56]. However, with the advent of Retina-Net [170] the one-stage detectors started to achieve comparable accuracy to that of two-stage detectors while maintaining very high detection speed. More specifically, the authors claimed that the extreme foreground-background class imbalance (which is even higher for images such as RSIs) encountered during the training of dense detectors is the central cause of lower performances. This is solved with the introduction of a new loss function named *focal loss* that forces detectors to put more focus on hard, misclassified examples during training.

Region proposal methods such as [168, 169, 171] are a fundamental preprocessing step that has been used in many CV tasks such as object recognition [49–51], IS [172, 173] and text detection [174] both in Fully Supervised and Weakly Supervised scenarios. Region proposal methods aim at extracting a certain number of regions of interest, i.e., those that may contain object instances, from the image. It can be accomplished in different ways, with the basic approach being a sliding window. More advanced and efficient proposals generation methods have been proposed, such as *Selective Search* [168], *Edge Boxes* [169] and *Multi-scale Combinatorial Grouping* (MCG) [171], that exploit low-level features like color and edges or low-level contour information (e.g., Structured Edge, Ultrametric Contour Map) as cues to produce object candidate windows. In general, these methods are built to have a high recall so that the generated candidates are highly likely to contain an object instance. However, these methods are highly time-consuming. To solve this issue, it is possible to either exploit approaches in which there is no region proposal generation step [40, 175] or directly integrate the region proposal generation and features extraction steps in the network using an *RPN* [51]. The latter exploits CNNs and can extract more relevant features for the areas of interest and speed up the process.

Weakly Supervised Object Detection (WSOD) [12] consists in performing OD exploiting coarse-grained annotations without the knowledge of ground truth BBs. The problem has been addressed by exploiting different types of annotations such as points [176, 177] and image-level annotations [178–182]. In both cases, the problem is ill-posed. In particular, knowing the image-level label of a training sample does not provide any type of information regarding the location of the objects in the image, while point annotations just provide coarse information on the objects' position. Given the nature of the problem, at the current time, image-level annotation-based methods struggle to reach the performances of FSOD methods. The gap between FS and WS approaches is reduced if point annotations are used, but still, some work needs to be done. In the context of WSOD, three main classes of approaches can be distinguished as reported in [12]:

- **Multiple Instance Learning (MIL) - based**
- **Class Activation Maps (CAM) - based**
- **Hybrid**

2.4.1. MIL-based Approaches

Most of the existing methods [178–183] for WSOD are based on MIL. In such cases, the image is viewed as a collection of potential instances of the object to be found. Typically, MIL-based weakly supervised object detectors follow a three steps pipeline:

1. **Proposal generation**: extract a certain number of regions of interest from the image that are highly likely to contain an object instance, exploiting a Region Proposal Method.
2. **Feature extraction**: compute a feature vector for each candidate region that contains the relevant information of that crop. Features can be handcrafted or

extracted by a CNN as in DL methods.

3. **Classification**: perform OD by recasting the problem as a MIL classification one.

The last step can be performed by exploiting the fact that it is natural to treat WSOD as a MIL problem. The MIL problem was first introduced in [184]. It is a classical weakly supervised learning problem. In the general MIL framework, the training set is composed of *bags* and each bag is associated to a *set of instances*. Each bag is labeled (*bag-level label*) as positive for a specific class if it contains at least one positive instance (*instance-level label*) of that class. On the other hand, a bag is labeled as negative for a class if it is associated only with negative instances of that class. Then the task to be solved can be one of the following:

- **Instance-level classification**: it consists in inferring the unknown instance labels from the known bag labels.
- **Bag-level classification**: it consists in inferring the unknown bag labels.

The WSOD task based on image-level labels only can be reformulated as an instance-level classification MIL problem. In this particular application, the bags are the *images*, whereas the set of instances to which an image (bag) is associated is the set of *feature vectors of region proposals*. Then, the bag-labels are *image-level labels* and the inferred instances-labels are the *BB-level labels*. For the training step, each image (or bag) is assigned a positive or negative label based only on the image-level label, i.e., the presence or absence of a specific category. Thus, an image can be represented as a positive bag for one category while a negative bag for another category not present inside such an image as shown in Figure 2.9. The aim is to infer instance-level labels for the proposals inside each image. Figure 2.9 illustrates the recasting of the WSOD problem as a MIL problem, while the following schema summarizes the relationship between MIL and WSOD:

$$\begin{aligned}
\textbf{MIL Problem} &\iff \textbf{WSOD Task}\\
\textit{Bags} &\iff \textit{Images}\\
\textit{Instances} &\iff \textit{Region Proposal feature vectors}\\
\textit{Bag-level labels} &\iff \textit{Image-level labels}\\
\textit{Unknown instance-level labels} &\iff \textit{Unknown BB labels}
\end{aligned}$$

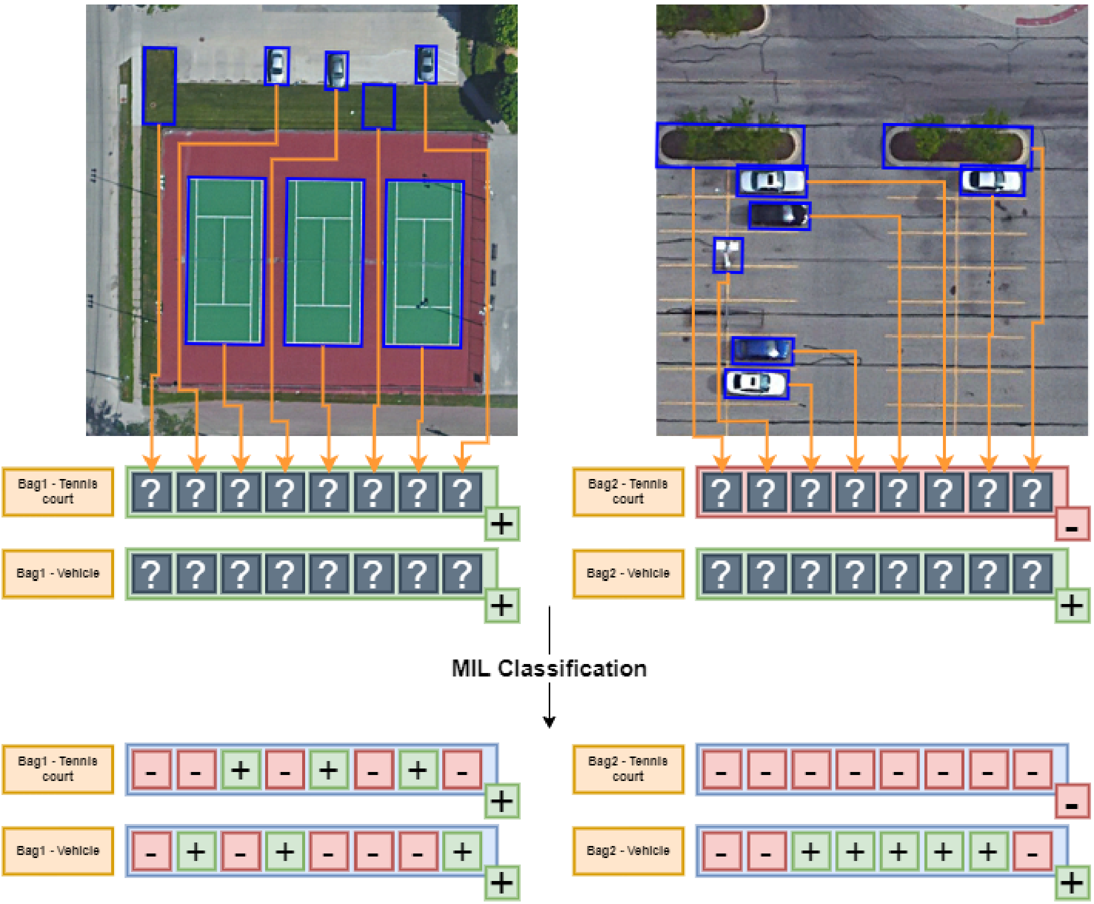


Figure 2.9: Illustration of a WSOD problem recast as an Instance-level classification MIL problem. The instances (BB proposals) are taken as input and Image-level labels are used as Bag-Labels. The considered categories are *Tennis court* and *Vehicle*. For Image (Bag) 1, both categories are present, this means that the bag is positive for both of the classes (as reported in the corner of the bag), while in Image (Bag) 2 only vehicles are present, so the bag is positive for the vehicle class and negative for the tennis court one. Initially, every proposal is associated with an unknown instance-level label, which is then learned during the MIL Classification, leading to the knowledge of the final BB-level labels of the images. The two images are taken from DIOR data set [34].

2.4.2. CAM-based Approaches

Another class of approaches for tackling WSOD is to formulate the problem as a localizable feature map learning problem. The idea comes from the fact that as observed in [185] every convolutional unit in the CNN is essentially an object detector that can locate the target object in the image. For instance, if the object appears in the upper left corner of the image, the upper left corner of the feature map after the convolutional layer will produce a greater response. These localization capabilities of CNNs have been further studied in other works such as [186, 187]. CAMs were introduced in [187] as a weighted activation map generated for an image (see Figure 2.10). A CAM helps to identify the region a CNN is looking at while classifying the image. Since no additional label related to the location is required to build CAMs, the objects do not have to be labeled manually and the localization is kind of learned for "free". Once the CAM has been obtained, BBs can be easily computed by post-processing it. Thus, several CAM-based methods have been developed in the last few years especially for the task of Weakly Supervised Object Localization (WSOL) [188–192], but also for WSOD [47, 175].

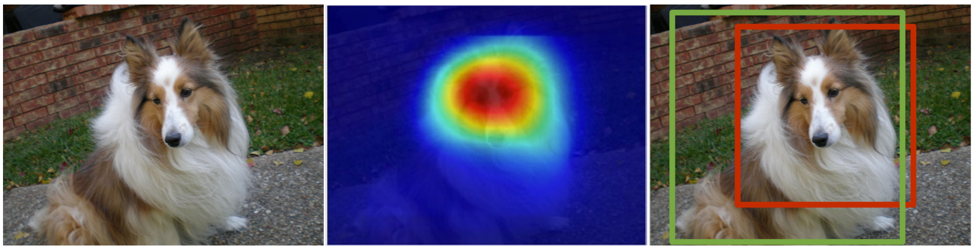


Figure 2.10: In the middle, example of CAM for the *dog* class obtained from the image on the left. On the right, the green BB is the ground truth, whereas the red BB is the one obtained by thresholding the CAM values.

2.4.3. MIL-based VS CAM-based Approaches

MIL-based and CAM-based approaches have advantages and disadvantages as analyzed by Shao et al. [12]. Firstly, MIL-based networks leverage region proposals methods such as Selective Search [168], Edge Boxes [169] or sliding windows to generate thousands of initial proposals, while CAM-based networks segment the activation map to obtain a proposal for each class (in the case of Object Localization). Therefore, a MIL-based method is usually better than a CAM-based method when detecting multiple instances with the same category in an image as is often the case for RSIs. Instead, a CAM-based method performs well when few big instances are present in the image (e.g., natural images). However, training and making inference using MIL-based networks are slower due to the presence of the region proposal generation step that is time-consuming and yield plenty of initial proposals most of which are not valid. Moreover, MIL-based approaches usually provide better overall performances but at the same time, the performance on each category is highly variant. On the other hand, CAM-based approaches are less widespread and effective than MIL-based approaches. They tend to be more stable in terms of performance over the classes. For this reason, it could be interesting to build hybrid approaches that exploit the advantages of both methods (see Section 2.4.4). Finally, as highlighted in [12] both MIL-based networks and CAM-based networks tend to suffer two main problems:

- **Partial coverage problem**: it may arise from the fact that the proposals surrounding the most discriminative part of an instance are likely to have the highest score. If proposals are selected solely based on the highest score, the detector will learn to focus only on the most discriminative parts and not the entire extent of an object (discriminative region problem). Another cause may derive from the use of proposal generation methods such as Selective Search [168], and Edge Boxes [169] whose proposals may not well cover the entire objects, severely hindering the performance of the detector (low-quality proposals).
- **Multiple-instance Problem**: accurately detecting all instances of the same category in an image is challenging given that object detectors [178, 179] tend to select the highest score proposal of each category as the positive proposal and ignore all the others.

2.4.4. Hybrid Approaches

Given the different characteristics of the two approaches and their corresponding advantages and disadvantages, an interesting option would be to study the possibility of exploiting the advantages of both approaches leading to a sort of hybrid approach. In particular, Cheng et al. [182] propose a MIL-based approach in which high-quality candidate proposals are generated by combining Selective Search [168] and Grad-CAM [193] since MIL-based methods usually require good quality candidates. Wang et al. [67] propose to use a WSOL method such as [187, 193] to generate pseudo-labels for each proposal. Then, during the training stage, low-quality proposals are effectively suppressed while high-quality proposals are highlighted.

2.4.5. WSOD in Natural Images

The problem of WSOD has been widely studied in the last few years since it allows the detection of objects without the need for ground truth BBs. The research focused on solving the most typical problems of WSOD such as the discriminative region problem and multiple-instance problem, accounting also for other critical factors such as speed [12].

In 2016, Bilen and Vedaldi [178] propose a milestone for WSOD starting from image-level labels named Weakly Supervised Deep Detection Network (*WSDDN*), based on

MIL. The core of WSDDN is a two streams network that aims to perform classification and localization respectively. Initially, the image and region proposals extracted using a region proposal algorithm, are fed into some convolutional layers with a Spatial Pyramid Pooling layer [194] to produce a fixed-size convolutional feature map for each proposal. Then, proposal feature maps are fed into two fully connected layers to produce proposal features. These features are then given as input to the two streams. The classification branch computes the class score of each proposal and the detection branch computes the contribution of each proposal to the image being classified as a certain class. These scores are then multiplied for each region and summed to obtain the final prediction score. Since only image-level labels are used for the training, the network tends to suffer the discriminative region problem.

To alleviate this weakness, Tang et al. (2017) [179] extends WSDDN [178], proposing a new architecture denominated *OICR*. OICR adds refinement branches to the basic instance classifier of WSDDN, that are designed to further predict the class scores for each proposal. Since the output of each branch is used as supervision for the next one, OICR can continue to learn so that a bigger part of the object can be covered. However, OICR only tackles the discriminative region problem, but still suffers as WSDDN, the multiple-instance problem since both methods select the highest score proposal of each category as the positive proposal and ignore all the others, hurting the discriminative power of the detector. However, since then, most MIL-based WSOD methods [65, 180–182, 195–200] are based on the structure of OICR [179] and aim to further alleviate the discriminative region problem and solve the multiple-instance one.

In 2018, Tang et al. [180] proposes an OICR-based method named Proposal Cluster Learning (*PCL*). As in OICR, the proposal features are branched into different streams (the basic instance classifier and the refinement ones). For each stream, proposal classification scores are obtained and *proposal clusters* are generated. Based on these proposal clusters, supervisions for the next stream are generated. Thus, PCL aims to treat each proposal cluster as a small bag to train refined instance classifiers. This is proved to force the network to cover the whole object or at worst larger parts of the object. One year later, Li et al. [195] shows the effectiveness of combining OD and SS tasks, introducing a segmentation-detection collaborative mechanism named *SDCN*. The proposed approach consists of a detection branch and a segmentation branch, which are responsible for detecting BBs and generating segmentation masks respectively. In this way, the detection and segmentation branches are optimized alternatively and promoted each other, resulting in a performance improvement.

In 2020, Chen et al. [181] proposes a Spatial Likelihood Voting (*SLV*) module that is fed with average classification scores of the refinement branches. The basic idea is to solve a multi-task problem in which the classification and localization tasks promote each other to improve the performances and alleviate the discriminative region problem.

Given that previous approaches do not take into consideration the consistency among different views of the same image, this could lead to labeling these views differently, hurting the performances. Thus, Huang et al. (2020) [200] proposes a Comprehensive Attention Self-Distillation (*CASD*) training approach for WSOD. CASD conducts consistent representation learning over input images under multiple transformations, which guarantees the feature consistency of related proposals of the same image under different transformations, solving the outlined issue. During the same year, Cheng et al. [182] proposes a MIL-based approach in which however proposal generation is improved using Grad-CAM [193] to produce better candidates. This is a very important point since the performances of the detectors are highly affected by the quality of the initial proposals. Usually, WSOD MIL-based methods train object detectors with the instances obtained by region proposal methods [168, 169]. However, given that the proposals generated by these algorithms, can not well fit the ground truth BBs, the learned object detectors may not well localize or even be unable to localize objects. This motivates the generation of higher-quality proposals.

Over the years researchers proposed other methods to cope with the discriminative region problem. Some of these methods make use of context information [201, 202], generative adversarial learning [203], gradient maps [204]. Furthermore, researchers focused on addressing the issue that MIL-based approaches tend to get stuck in local minima due to the introduced false positive examples. To solve this issue, some of the proposed methods are based on modeling the uncertainty in the location of the objects [205], continuation optimization [206] and zigzag learning strategy [207].

Besides the MIL-based framework which is the most widely used approach, other methods based on CAMs [188–192] have been developed. However, these approaches are widely used to solve the task of WSOL which aims to detect only one instance in an image. The main reason is that CAMs usually provide better results when images contain a few large objects. Shao et al. [12] provide a well-structured report of the recent advances in this area of research.

2.4.6. WSOD in Remote Sensing Images

The works analyzed in the previous subsection have been trained and tested on natural images. However, the application of WSOD methods is not limited to this type of image.

WSOD methods that exploit image-level labels are also widely used with RSIs to address problems such as vehicle detection [167] and aircraft detection [208, 209]. Performing Remote Sensing Weakly Supervised Object Detection (RSWSOD) is a harder task with respect to WSOD in the natural images domain since both the partial coverage and multiple-instance are still present, but at the same time, also additional challenges arise from the RSIs characteristics (see Section 2.1.2):

- **Density problem**: in RSIs, there are often dense groups of instances belonging to the same category. Models usually have difficulties in accurately detecting and distinguishing all the instances (see Figure 2.3).
- **Generalization problem**: the high intra-class diversity in RSIs induces generalization problems mainly due to three factors:
  - **Multi-scale**: objects may have varying sizes, and their representation strongly depends on the image resolution and Ground Sample Distance (GSD).
  - **Orientation variation**: instances present arbitrary orientations and may require the use of methods generating Oriented BBs instead of the classical Horizontal BBs.
  - **Difficulty**: in general, RSIs show varying detection difficulty based on the background and instances complexities. Multiple instances of the same category may be more or less difficult to detect in the same image.

Before the advent of Deep Learning (DL), most object detectors were based on *Support Vector Machines* (SVMs). The workflow behind these methods is to start by producing candidate proposals exploiting either a *Sliding Window* (SW) [165, 210, 211] or *Saliency-based self-adaptive Segmentation* (Sb-SaS) [37, 42, 212, 213] approach. SW generates proposals by sliding, on the entire image, multiple BBs with different scales while Sb-SaS produce saliency maps that measure the uniqueness of each pixel in the image and exploit a multi-threshold segmentation mechanism to produce BBs, dealing with the variation of the target size and the resolution of the RSIs. Each proposal is characterized using a set of low- and middle-level features derived using methods such as SIFT [214] and HOG [215]. The extracted features can be further manipulated to produce high-level ones. Then, sets of positive and negative candidates are chosen to perform Training Set Initialization (TSI) and finally Target Detector Learning (TDL) is performed. The detector training procedure is composed of two steps: 1) training of the detector and 2) update of the training set (modifying the positive and negative candidates). These steps are repeated until a stopping condition is met. In this work, this type of approach is referred to as

TSI+TDL.

Later on, thanks to the advancements in the DL field and the development of more powerful high-level feature extractors and CV architectures, researchers moved towards MIL-based [65, 196, 198, 199] and CAM-based [47, 175, 209] methods that could be more efficient while providing better performance, with the same distinction present in the natural images domain. However, the SOTA methods for natural images must be adapted to face the challenges induced by the usage of RSIs. In this direction, several approaches have been proposed to reduce the inference of the background and allow the distinction of adjacent instances.

The first attempts to apply WS techniques to aerial images to perform single-object detection are TSI+TDL-based approaches, performed by Zhang et al. [37, 42] in 2014. These first works aimed to reduce the effort needed for FS methods by proposing working solutions for WSL on RSIs. The idea is to mine positive and negative samples to initialize the training set and then exploit an iterative training scheme to refine the detector and update the training set using a WS SVM. Inspired by this work, Han et al. [210] proposes a probabilistic approach using the Bayesian rule [216] to jointly integrate saliency, intra-class compactness, and inter-class separability to better initialize the training set, information that was not considered by the previous works. This work also highlights the limitations of low-level and mid-level feature extractors that are not powerful enough to effectively describe objects in RSIs due to the influence of the cluttered background and proposes to use a *Deep Boltzmann Machine* [217] to extract high-level features. All these methods focus on the problem of single-object detection. Cheng et al. [211] attempted to employ a Collection of Part Detectors (COPD) [43] composed of a set of WS SVM detectors to adapt previous works to multi-object detection.

With the advent of CNNs [53], both WSOD and RSWSOD methods started to benefit from the powerful feature extraction capabilities of deep architectures. In 2015, Zhou et al. [212, 213] propose to use transferred deep features and negative bootstrapping to make the detector converge more stably and faster by exploiting the most discriminative training samples. To avoid the use of time-consuming methods for proposals generation, in 2016, Zhang et al. [208] proposed the use of a coupled CNN that integrates an RPN [51].

The introduction of WSDDN [178] and OICR [179] was a milestone for WSOD in natural images. Of course, it also had a huge influence on the Remote Sensing community. However, the direct application of these methods to solve tasks in the RS domain results in a severe performance drop. For this reason, many researchers focused on solving the

RSWSOD problem by improving these techniques by adding new modules that could overcome RSIs challenges.

In 2017, Cao et al. [165] exploit MIL and density estimation to predict vehicles' locations starting from region-level labels. One year later, Sheng et al. [166] propose *MIRN*, a MIL-based approach that tries to leverage the count information and an online labeling and refinement strategy, inspired by OICR, to perform vehicle detection, solving the multiple-instance problem. During the same year, Li et al. [218] proposes a Siamese network to overcome the fact that existing methods tend to take scenes as isolated ones and ignore the mutual cues between scene pairs when optimizing deep networks. Moreover, a multi-scale scene-sliding-voting strategy is implemented to produce the CAM allowing to solve the multi-scale problem. The authors further propose different methods for thresholding the CAM and observe that the detection results for each class have a strong dependence on the chosen thresholding method. In 2019, Ji et al. [209] propose a method to reduce the false detection rate that affects many aircraft detectors producing a more accurate attention map, while Aygunes et al. [219] modify WSDDN architecture to adapt it for the task of WS Fine-Grained Object Recognition for tree species classification, which is even more challenging than traditional RSWSOD given the very low inter-class variance. The same authors further improve the work, addressing the same task under the presence of multiple sources [220]. In this case, WSDDN is used to perform RSWSOD using the multispectral image and LiDAR data, while the RGB image (assumed to have no location uncertainty) is exploited as a reference to aid data fusion, which is a critical step in multi-source scenarios.

In 2020, Chen et al. [221] proposes a novel MIL-based object detector based on a neural network called Full-Coverage Collaborative Network (*FCC-Net*). Hybrid dilated convolutions and multi-level pooling techniques are combined to fuse multi-scale feature maps allowing the detection of various-sized objects which represent a major problem in RSIs. Moreover, the authors show that iteratively training a WS detector and an FS one exploiting the output of the WS detector as pseudo-ground truth can improve the performances.

Feng et al. [196] propose a new MIL-based approach based on OICR [179] called *PCIR*. PCIR tries to solve the discriminative region problem by exploiting a context-based strategy to divert the focus of the detection network from a local distinct part to the whole object and further to other potential instances, also addressing the multiple-instance problem. A progressive proposal self-pruning algorithm is further designed to mitigate the influence of complex background typical of RSIs by dynamically rejecting the negative training proposals. During the same year, Feng et al. [198] propose *TCANet*, a

MIL-based architecture that aims to exploit context information as in the case of PCIR to alleviate the problem of the discriminative region. At the same time, appearance similarity is considered to learn instance-level discriminative cues that allow easily distinguishing instances appearing in adjacent locations, thus solving the density problem. In their following work [199], the authors propose a method based on the TCANet architecture [198] named *SAENet*. Adversarial Dropout–Activation (ADA) blocks are used to capture the whole object and a self-supervised transformation equivariance mechanism is used to enforce consistency within an instance and its augmentations. This is a major contribution since previous methods didn't consider this issue. Thanks to the proposed method, an object instance in an image and the same instance in the augmented one are equivalently classified. If this is not constrained, they may be classified differently and this could damage the detector's performance. This fact was already highlighted in WSOD by Huang et al. [200].

Another important aspect of training a good detector is the quality of the proposals, as already previously outlined [182]. Yao et al. [197] highlight that besides the importance of proposals' quality, most approaches often fail to provide high-quality initial samples to the detectors, making it difficult to obtain optimal object detectors. To address this issue, a dynamic curriculum learning strategy [222] is proposed to progressively learn the object detectors by feeding training images with increasing difficulty. The difficulty of localizing objects in images is assessed by employing an entropy-based criterion. Then, training images are fed to the detector in ascending order of difficulty, leading to an improvement in its capabilities. This intuition was dictated by the recent advances in natural images WSOD [223, 224].

The importance of the quality of image proposals is further studied by Wang et al. [67] that propose an interesting MIL-based approach inspired by PCL [180] to perform object detection. The key innovation of the paper is the proposal generation step: a novel pseudo-label generation (PLG) algorithm is developed combining Selective Search [168] with the information provided by a CAM-based weakly supervised localization model [225]. This way, by intersecting the results of selective search with those of the WSOL method, low-quality proposals can be effectively suppressed. Moreover, recently, Cheng et al. [226] propose an RPN based on the objectness confidence to generate high-quality proposals. The authors show that using the proposed RPN in place of standard techniques (e.g., Selective Search [168]) can improve the performance of previous MIL-based methods such as OICR [179] and MELM [227].

In the meantime, Wu et al. [47] proposes an effective way of using a CAM-based approach for aircraft detection. In their subsequent work [175], the authors propose a

CAM-based approach that fuses the information extracted from shallow CAMs (*SCAMs*) and deep CAMs (*DCAMs*), reducing the performance gap between MIL-based and CAM-based approaches. In particular, the work highlights the fact that SCAMs are useful to localize objects while DCAMs are useful to be able to classify instances correctly. Moreover, Divergent Activation [189] and similarity modules are used to identify more objects and find densely-distributed objects. Furthermore, the authors show that the proposed method performs similarly to PCIR [196] with the difference that PCIR like many MIL-based methods, tends to perform very well for some categories and poorly for others while the proposed CAM-based approach is more balanced.

In 2022, Qian et al. [65], propose a MIL-based approach based on OICR [179] with modified losses to account for completeness and difficulty. The authors are the first to highlight that an imbalance between easy and hard samples causes the network not to learn how to correctly detect objects in the few available hard samples. This method makes use of segmentation masks produced by the WSSS algorithm proposed in [228] to determine the completeness and difficulty of each sample. Completeness is used to state whether a proposal covers the entire object, while difficulty evaluates how difficult it is that a proposal can be identified correctly. The authors also state that using a more robust WSSS method can increase the accuracy.

Over the years, other works have been developed to address more specific tasks. For instance, Du et al. [229] propose an RSWSOD method based on the TSI+TDL framework and image-level labels to detect objects in Synthetic Aperture Radar (SAR) images. Li et al. (2021) [230] proposes an RPN for geospatial applications that consider Tobler's First Law of geography, stating that *Everything is related to everything else, but near things are more related than distant things.* The idea is to convert the 2D object detection problem into a 1D temporal classification problem. The method is applied for terrain feature detection. Berg et al. [231] exploited an anomaly-detection mechanism to detect marine animals from aerial images. By training this model using images not containing marine animals, the model will then be able to detect animals as anomalies. Finally, Yang et al. [40] addressed the task of ship detection by exploiting an image transformer [232] called *PistonNet* which showed impressive generalization capabilities for generic object detection, leading the way to transformer-based solutions.

The works reported in this section show that MIL is the most widely used framework to solve RSWSOD. Several of the reported methods are applied to generic scenarios, even though some specific applications have been studied (e.g., vehicle detection, tree species classification), and it is possible to compare their performances on the most popular data sets such as NWPU VHR-10.v2 [44], DIOR [34] and Google Earth [37].

Table 2.2: Overall results of some WSOD methods on the main data sets. Only the methods with available results on DIOR, NWPU VHR-10.v2 or Google Earth are reported.

Name	Approach	Year	NWPU VHR-10.v2		DIOR		Google Earth
			mAP	CorLoc	mAP	CorLoc	AP
Zhang et al. [37]	TSI + TDL	2014	-	-	-	-	54.18%
Han et al. [210]	TSI + TDL	2014	-	-	-	-	60.16%
Zhang et al. [42]	TSI + TDL	2014	-	-	-	-	66.42%
Zhou et al. [212]	TSI + TDL	2015	-	-	-	-	75.58%
Zhou et al. [213]	TSI + TDL	2016	-	-	-	-	**76.26%**
FCC-Net [221]	MIL	2020	-	-	18.30%	41.70%	-
DCL [197]	MIL	2020	52.11%	69.65%	20.19%	42.23%	-
PCIR [196]	MIL	2020	54.97%	71.87%	24.92%	46.12%	-
AlexNet-WSL [47]	CAM	2020	-	-	18.78%	-	-
TCANet [198]	MIL	2020	58.82%	72.76%	25.82%	48.41%	-
Wang et al. [67]	MIL + CAM	2021	53.60%	61.50%			-
SDA-RSOD [175]	CAM	2022	-	-	24.11%	-	-
MIGL [233]	MIL	2021	55.95%	70.16%	25.11%	46.80%	-
SAENet [199]	MIL	2021	60.72%	73.46%	27.10%	49.42%	-
SPG+MELM [226]	MIL	2022	**62.80%**	73.41%	25.77%	48.30%	-
Qian et al. [65]	MIL	2022	61.49%	**73.68%**	**27.52%**	**49.92%**	-

This comparison, reported in Table 2.2, shows that CAM-based methods have been less evaluated on these three challenging data sets. This may happen because, most of the time, CAM-based approaches are used for specific tasks such as aircraft detection, while MIL methods are more frequently applied to generic scenarios. Another factor influencing this trend is that CAMs work well when there are few large instances in the image [12]. For this reason, MIL-based approaches seem more effective than CAM-based ones in RSIs, where multiple instances are present. However, recently, this performance gap has been reduced by the work of Wu et al. [175].

It is important to note that the overall performance of the methods increases over the years independently of the data set, demonstrating the effectiveness of the novel proposed methods.

Even though there has been a great improvement in the performances of WSOD methods based on image-level labels, there is still a consistent gap w.r.t. FSOD approaches such as YOLOv5 [234], as shown in Table 2.2. Because of that, the problem of WSOD in RSIs has also been addressed by exploiting finer-grained annotations such as point labels [176]. The idea behind the usage of this type of annotations is that they are far cheaper than the BB annotations [176] and they allow to greatly reduce the performance gap between WS and FS approaches.

An in-depth analysis of State-of-the-art RSWSOD approaches was performed and published in a systematic review [13].

2.5. Weakly Supervised Instance Segmentation

Instance Segmentation [24] is another very important and studied task in the field of CV. IS allows assigning a label to every pixel in the image, discriminating different instances of the same class. This is the difference between SS and IS. Several solutions for IS tasks have been developed in the last few years to address problems such as ship detection [38] and pedestrian detection [235].

In an FS scenario, it would be necessary to annotate the data set with a mask for each specific instance (set of pixels corresponding to each instance) and the class label for each mask. According to Hafiz and Bhat [24], there are four main families of approaches that can be considered in an FS scenario:

- **Classification of mask proposals**: mask proposals are generated using methods such as Selective Search [168], and this is followed by a classification of the generated proposals.
- **Detection followed by segmentation**: the generation of the segmentation masks starts from bounding boxes and is followed by object-box segmentation.
- **Labelling pixels followed by clustering**: the usage of techniques for SS is followed by the application of clustering algorithms to generate instance masks.
- **Dense sliding-window methods**: dense sliding-window techniques are used in CNNs for mask proposal generation.

Mask R-CNN [52] is probably the most widely used architecture for IS and it belongs to the *Detection followed by Segmentation* approaches. Other main approaches are reported and well categorized in the survey by Hazif and Bhat [24].

Weakly Supervised Instance Segmentation (WSIS) consists in performing IS exploiting only coarse-grained labels such as image-level labels [172, 173, 228, 236, 237]. It is an ill-posed problem since knowing the image-level label does not provide any type of information regarding the location and the label of the instances of the objects. It is also possible to use BBs as coarse-grained labels [238–240]. In this way, the information regarding the location of the object is known but no insight is given about the pixels representing the specific instance. WSIS with only image-level annotations is the most difficult problem to be addressed among those analyzed so far, but, at the same time, annotating a data set for Fully Supervised IS would be very time-consuming.

2.5.1. WSIS in Natural Images

WSIS methods based on image-level labels only [172, 173, 228, 236, 237] are still a relatively new area of research, and most of them are focused on the usage of CAMs [187] and their variations. There exist other methods for WSIS based on BB-labels [238, 240].

In 2018, Zhou et al. [172] propose the first approach to address the WSIS task starting from image-level labels. The basic idea is to exploit the fact that CAMs show peaks in regions corresponding to instances' locations. For this reason, a peak stimulation layer is used to better localize instances and produce Peak Response Maps (PRMs). Finally, by combining instance-aware cues from PRMs, class-aware cues from CAMs, and spatial continuity priors from object proposals generated by off-the-shelf methods (e.g., Multiscale Combinatorial Grouping [171]), instance masks can be produced. Since then, most approaches are based on this model and aim to solve its tendency to focus on the most discriminative part of the object, dictated by the use of CAMs. In particular, Laradji et al. [241] builds on PRM by using its output pseudo masks to train an FS method, namely, Mask R-CNN showing the effectiveness of this pipeline.

One year later, Ahn et al. [228] propose IRNet to generate pseudo instance segmentation labels of training images and use them for training an FS model. The method is proposal-free and focuses on class-equivalence relations between a pair of pixels and represents instance-level information using their displacement field. Thus, this method has the advantage of not requiring the use of off-the-shelf methods for the generation of mask proposals as instead is required by almost all the other methods. During the same year, Ge et al. [236] propose a method named *Label-PEnet* that progressively transforms image-level labels to pixel-wise labels in a coarse-to-fine manner. The method performs four tasks sequentially: multi-label classification, object detection, instance refinement, and instance segmentation. The cascaded pipeline is trained alternatively with a *curriculum learning* strategy [222] that generalizes labels from high-level images to low-level pixels gradually with increasing accuracy. The curriculum learning strategy, already used also in OD [197, 224] allows the network to start learning an easy task and when the model has started to learn, introduce a more complex task, improving the performances.

In 2020, Liu et al. [242] integrate the useful information of all training images into a large knowledge graph and explore the information in this graph to bridge the image-level keywords and corresponding semantic instances. In this way, the method takes into consideration not only the intrinsic properties of each image but also the overall data distribution of the training database, so that it breaks the limitations of CAMs on Weakly Supervised Semantic Segmentation (WSSS). During the same year, Arun et al. [173]

proposed a method that, unlike previous approaches, explicitly models the uncertainty in the pseudo-label generation process using a conditional distribution, given the fact that the proposals generated by off-the-shelf methods [168, 169, 171] are not accurate enough. Furthermore, it represents the instance segmentation model as an annotation-agnostic prediction distribution. This representation allows to define a joint probabilistic learning objective that minimizes the dissimilarity between the two distributions.

More recently, Kim et al. [237] propose a method named *BESTIE* that performs WSIS by transferring the knowledge from WSSS and without the need for mask proposals generated by off-the-shelf-methods. Moreover, the authors highlight an important issue in WSIS called *semantic drift* that has never been considered before. The problem is due to the missed instances in pseudo-instance masks categorized as background. This semantic drift causes confusion between background and instance in training and consequently degrades the segmentation performances.

2.5.2. WSIS in Remote Sensing Images

To the best of our knowledge, there is no research on WSIS that makes use of image-level labels only in the specific scenario of RSIs. Even though box-supervised instance segmentation has recently attracted attention, still little relevance is received in the aerial image domain. In this direction, Li et al. [239] propose an instance segmentation method for aerial images based on BBs, while Chen et al. [66] present a pipeline of hybrid supervision for instance segmentation also in the case of aerial images. In particular, a segmentation model to generate accurate pseudo-pixels-wise labels from real-world aerial images is implemented which only needs a small portion of pixel-wise labels for training. All the remaining work is done exploiting BB-labeled samples.

2.6. Illegal Landfills Detection

The possibility of capturing high-quality RSIs employing aerial devices has led to the possibility of extracting the highly valuable information that is hidden in this data to perform several important tasks in different fields, ranging from building extraction [64, 243] to change detection [112, 119, 153], vehicle detection [166, 167], marine animals detection [231] and asbestos coverings detection [244]. Among the wide range of different fields that can benefit from this, environmental monitoring is of great importance especially nowadays. In this field, Illegal landfills detection [8] and solid waste management [245] are quite relevant.

Illegal landfills are waste disposals on non-authorized sites or even authorized ones containing waste types or waste amounts that exceed the limits of the authorization obtained. The demographic increase has a considerable impact on the waste generation [2] and this phenomenon could lead to the birth of new illegal landfills. Of course, each country has its own legislation for what concerns the treatment of garbage to prevent impacts on the environment and society. In fact, illegal landfills can be a source of hazards for both the environment and people [3–5, 246, 247]. More precisely, waste is often set on fire, for instance, to eliminate evidence of dangerous materials. However, burying waste can result in the release of toxic fumes that can put public health at risk [248]. At the same time, if waste treatment is not performed carefully, as is often the case for unauthorized landfills, the release of leachate in the environment can pollute water sources. In the long-term, this could lead to relevant damages such as the increase in cancer incidence [7]. In fact, emissions and toxicological hazards from illegal dump sites can be extremely high compared to regulated landfills [3]. For this reason, detecting illegal disposal sites on time is crucial to reduce the impacts on the environment and society. In some cases, services are provided to citizens so that they can report illegal waste disposals [249]. However, merely relying on citizens is neither efficient nor completely reliable and thus, it is possible to exploit solutions that can automatically detect illicit sites exploiting the information that can be, for instance, provided by aerial images, without the need of performing a full on-site inspection of the territory. For this reason, the problem of illegal landfills detection has been addressed by several researchers over the years.

In the following, a review of the most important characteristics of illegal waste disposals and approaches for their detection is provided.

2.6.1. Illegal Landfills Characteristics

Landfills are arrangements of different types of materials. When observed from overhead, waste dumps appear as complex stacks of objects with varying shapes, sizes, and orientations. As reported in [8] in many cases, a waste dump area contains sparse debris, pallets, containers, and car carcasses. However, these areas are typically isolated places that can be reached by secluded roads. Moreover, according to [250], another important index of the presence of illegal landfills is the fact that the area is characterized by stressed vegetation.

As already outlined, the objects and materials that are typically deposited in dumping sites widely vary in their appearance. Examples of such materials include organic waste, plastics, glass, metal, paper, wood, textiles, tires, bulky waste, electronics, and hazardous waste [251]. Figure 2.11 shows a few examples of illegal landfill sites.

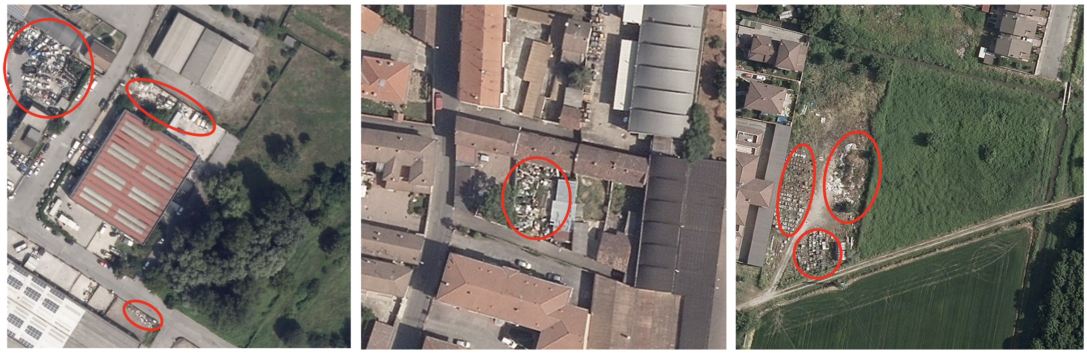


Figure 2.11: Examples of the presence of waste in potentially illegal sites. Red circles indicate suspicious objects. In all images accumulations of various materials and scattered waste are present. In the first image on the left, some car carcasses are abandoned at the sides of the shed. The image is taken from [8].

Even though during recent years a few techniques have been developed to automatically detect suspicious sites starting from RSIs [8, 16, 17], most of the time manual photo interpretation remains the predominant technique. This means that experts are needed to analyze the images to assess whether a suspicious site is present or not and, if possible, distinguish among the different types of objects that are present in the scene. This hinders the possibility of rapidly detecting sites and of analyzing a vast territory. However, recent DL techniques have been successfully developed to solve specific tasks such as asbestos coverings detection [252, 253] exploiting RSIs. For this reason, these technologies seem promising also for illegal landfills detection. However, it is crucial to take into consideration the challenges introduced by RSIs described in Section 2.2. As a matter of fact, illegal landfills aerial images are characterized by the following peculiarities:

- **Intra-class diversity**: the type of waste present in illegal landfills is varying (e.g., plastics, glass, tires, building material, car carcasses). The same holds for the distribution of the waste which can be for instance scattered or collected in dumpsters, as well as the different geographical areas in which the landfill is placed (e.g., rural or urban).

- **Inter-class similarity**: in some cases, areas in which there is no illegal landfill are quite similar to illegal landfills. This is for instance the case of industrial districts, legal landfills, and cemeteries.

- **Variable scale**: illegal landfills usually possess a varying scale that can depend on the distribution of the objects in the area (clustered or scattered). The same holds also for the different types of objects that can be present in waste disposal sites (e.g., car carcasses, tires).

- **Multiple types of objects**: a single waste disposal site captured by an image usually contains multiple types of objects that are not easily distinguishable from overhead.

Besides these characteristics which are typical of RSIs, other important factors should be taken into account for the specific scenario of illegal landfills. In particular, collecting ground truth samples is complex due to the sensitivity of the domain, which may be subjected to restrictions concerning the public release of the data sets. For this reason, the number of available data may be limited, and WS approaches such as those reported in the previous sections may be considered.

At the same time, another relevant challenge is concerned with the limitation of using training and testing data from the same geographical domain which may affect the performances if the data set is not split properly. This aspect needs to be carefully taken into account when building a new model.

2.6.2. State-of-the-art in Illegal Landfills Detection

The problem of illegal landfills detection covers a crucial role in environmental monitoring processes. Over the years, the problem has been addressed by exploiting different types of data such as on-site images and RSIs. Approaches for identifying and mapping illegal landfills, can, for instance, be developed exploiting the spectral signature of materials and of the contaminated surrounding vegetation [254].

The first studies concerning the detection of illicit waste disposals were based on the human interpretation of aerial images. More specifically, in 1974, Garofalo et al.[255] propose the use of aerial images to determine the spatial distribution of waste producers and waste quantities. A few years later, Erb et al. [256] show the importance of employing historical aerial photos to document landfills' existence, location, extent, and possible nature.

In 2008, Silvestri et al. [257] introduce a method that exploits remotely sensed information and a Geographic Information System (GIS) to identify unknown landfills over large areas in the north of Italy. The method is based on the spectral signatures of the above-landfill-growing vegetation. Information such as contamination effects on the radiometric properties of vegetation, the position of the road network, population density, and historical aerial photographs have been used to define and then filter numerous candidate sites that are most likely to host waste materials. Moreover, the authors highlight the importance of integrating GIS and RS information. Finally, they find that stressed vegetation is present in all the illegal landfills and conclude that it represents an important index of the presence of illicit waste sites. A similar approach is presented around a decade later by Gill et al. [258], where it is shown that waste decomposition is always associated with the production of warmer landfill gas that can be detected via the Landsat thermal sensor.

In 2017, Selani et al. [259] compare modeled, satellite, and collected data using GIS methods to determine the most accurate estimate of detecting illegal dumping. In particular, they classify WorldView2 high-resolution 8-band multispectral images into six categories, two of which refer to waste (building rubble, domestic dump). The authors obtained 85.16% accuracy using an SVM on the validation set (30% of total images) starting from a total of 610 observations. Moreover, the authors highlight the fact that not all the bands have the same importance for classifying different categories.

One year later, Angelino et al. [260] exploit satellite images to identify illegal landfills in the south of Italy relying on photo-interpretation performed by experts. However, to reduce this time-consuming task, a multi-feature detection algorithm is implemented. The detected sites have been then classified according to the type, state, location, and activity of the dumps. Moreover, a multi-temporal analysis has been performed employing multi-temporal satellite images, allowing to control the evolution of the phenomenon. This allows both finding new illegal spills and following the evolution (in terms of extension and persistence) of landfills already found in the past.

With the advent of DL, new possibilities to make illegal landfills detection more efficient became available. However, to the best of our knowledge, only a few approaches have been developed for the specific domain of illegal landfills. In 2019, Abdukhamet et al. [16], propose a modified version of RetinaNet [261] using DenseNet [262] as a backbone to formulate the detection of illegal landfills as an OD task. The authors obtained 84.7% average precision using an Intersection Over Union (IoU) of 0.3. The employed data set contains more than 2,000 images of the Shangai district annotated with bounding boxes framing the garbage. Moreover, the authors highlight the importance of data augmentations to enlarge the data set and show that the obtained performance depends on the size of the images, given that different amount of context is then provided during training according to this size.

In 2021, Youme et al. [17] present a DL-based automatic solution for the detection of clandestine waste dumps using Unmanned Aerial Vehicle (UAV) images in the Saint

Louis area of Senegal, West Africa. The problem is formulated as an FS multi-scale OD task, solved with the use of a Single Shot Detector. The employed data set comprises 5,000 annotated images, 10% of which were reserved for testing. The results show that the model recognizes well the areas concerned, but presents difficulties in some areas lacking clear ground truths. The model generated many false positives given to confusion with non-waste objects (e.g., trees). Devesa and Brust [263] implemented a CNN model based on U-Net architecture [264] to detect illegal landfills through an FS segmentation task. The model's performance in predicting obtained an IoU of 0.6304.

During the same year, Torres et al. [8] present a new DL solution to perform illegal landfills detection exploiting RSIs. In this case, the authors formulate the problem as a multi-scale scene classification task. More importantly, given the fact that annotating images is highly time-consuming and requires expert knowledge, the authors only make use of image-level labels indicating the presence or absence of an illicit waste disposal site. A data set of around 3,000 images (20 cm resolution per pixel) is thus created with the help of expert photo interpreters. To solve the task, the authors employ a Resnet50 architecture [21] modified to also integrate a Feature Pyramid Network (*FPN*) [22] to account for the presence of multi-scale objects. In this way, the classifier is trained to predict the presence or absence of an illegal landfill, reaching 88.6% precision with an 87.7% of recall on a test set and an Expected Calibration Error (*ECE*) of 7.01. This last metric allows indicating how well the probability estimates provided by the model can be interpreted as correctness likelihood. In a well-calibrated classifier of all the samples that are predicted with a probability estimate of, say, 0.8, around 80% should belong to the positive class [265]. The authors state that in the case of illegal landfills detection, it is particularly important that the model output reflects the actual underlying probability of the positive class to support the decision to inspect a suspicious site (which is fundamental to reduce the number of sites that require on-site inspection). Furthermore, the authors provide a qualitative evaluation of the results. A visual inspection of the results helps to understand the behavior of the model and to understand which are the objects on which the model tends to focus more. This is achieved by exploiting multi-scale CAMs [225] which proved that the model tends to focus on the same aspects considered by the human experts.

The same authors in another work [266] present a comparative study about the effectiveness of CAMs as a tool for explaining how a CNN-based classifier recognizes suspicious objects potentially denoting the presence of a waste dump in aerial images. The authors apply four classification CNNs: ResNet50 [21] is used as a baseline, while the other three architectures are equipped with alternative attention mechanisms (Squeeze and Excitation

(SE) [267], ECA-Net [268] and CBAM [269]). The quantitative evaluation is performed based on IoU, on a test data set that comprises 596 images annotated with 3,411 segmentation masks surrounding specific waste objects. The comparison shows that the inclusion of attention methods in different variants improves the performances with respect to the baseline and that, among these variants, ECA-Net shows the greatest capacity of highlighting the distinct components of the image that represent the waste dump location.

The authors further collected more data into a data set (AerialWaste) [18] and repeated the experiments presented in the work [8], obtaining 81.9% precision and 79.5% recall on the test set.

In 2022, Karimi et al. [270] showed an interesting approach for detecting illegal dump sites using night satellite imagery and various remote sensing indices. During the same year, Djidelija et al. [271] try to discover the value of the scale parameter that gives the best results in detecting illegal landfills in a segmentation approach. Moreover, despite using the optimal scale parameter, the authors recommend considering detected sites as potential until verification is done. Finally, they also state that to obtain accurate results, high-resolution satellite images are necessary.

Rajkumar et al. [272] present a very high-resolution landfill data set created from satellite images for illegal landfills and demonstrate that by applying suitable DL FS segmentation methods, landfills can be detected even with constrained and limited data set.

Given the lack of annotated data, Padubidri et al. [273] develop a method for detection and reporting of illegal dumping sites from high-resolution airborne images based on DL, training the architecture on synthetic data. The authors evaluate the use of two different architectures, i.e., a basic CNN classification model with three hidden layers and a deeper model with residual blocks [21]. The image patches were given as input to the DL classification models and a class for each patch was predicted indicating the presence or absence of dumps. At test time, a real-world data set is used and coordinates are output to indicate the location of the potential waste disposal sites. The residual model obtained 97% precision and 92% recall.

RSIs are not the only source of information that could be exploited to perform illegal landfills detection and recognition. GIS is used in many methods [257, 259, 274, 275], combined with RSIs or with Multi-criteria evaluation methods (MCE). One of the ideas of GIS-MCE methods is to assess the probability of occurrence of illegal landfills and, based on the probability values calculated for each candidate site, to construct a priority list.

A complementary problem to illegal landfills detection is Street-level garbage detection, addressed in several works [276–281]. In most of these works, SOTA DL architectures [21, 51, 282] were successfully applied to classify waste types at street level, showing the capability of the methods to learn features that characterize waste objects viewed up close. However, to the best of our knowledge, no study has been performed on waste type classification starting from RSIs.

3 | Data Set, Architecture, and Methods

In this chapter, information concerning the data set, architecture, and methods designed for the various experiments, are provided. Specifically, Section 3.1 provides an in-depth analysis of the employed data set, its main characteristics, and challenges that may arise during the experiments; Section 3.2 analyses the exploited architecture, as well as the loss function and mechanism to generate CAMs; finally, Section 3.3 describes the approaches designed to face the previously identified challenges as well as the techniques proposed to try to enhance the discriminative power of the architecture.

3.1. Data Set

This work assesses the performance of multi-label fine-grained classification in the specific case of illegal landfills. More specifically, the aim is to build and evaluate a model that is able to distinguish among different entities that are present in illegal landfills. Most of the time, many items are present in the same waste disposal site. In particular, the capability to distinguish between the different illegal landfills is a fundamental requirement for the development of WSL approaches for the detection of illegal landfills, such as WSOD or WSIS.

To tackle the considered multi-label classification task, the AerialWaste [1] data set [18] is used. This data set has been built for the specific task of discovering illegal landfills. It consists of 10,434 RGB images from three different sources (AGEA, WorldView3, and Google Earth). The size and resolution of each image depend on the source the image was collected from. More specifically, the images provided by the Italian Agriculture Development Agency (AGEA) possess a spatial resolution of around 20 cm Ground Sample Distance (GSD) and a size of 1,000×1,000 pixels; images provided by WorldView3 are high-resolution images collected by a commercial satellite that possess a spatial resolution of around 30 cm GSD and a size of 700x700 pixels; finally, the images provided by Google

[1] https://aerialwaste.org/

Earth are images downloaded using the Google API [2] that possess a spatial resolution of around 50 cm GSD and a size of 1000x1000 pixels. The original data set is already split into training (75%) and testing (25%).

The images are provided with different annotation types:

- **Binary labels**: each image is annotated with a label that indicates whether the image contains an illegal landfill or not. An image that contains no landfills is regarded as a negative sample, while an image that contains at least an illegal landfill is a positive sample. Positive samples are 3,478, while the number of negatives is 6,956. The training set contains 2,612 positive and 5,217 negative samples, while the test set contains 866 positives and 1,739 negatives. Most of the time, in positive samples, the illegal landfills are located in the middle area of the image.
- **Multi-class multi-label**: a subset of images (715) is fine-grained annotated based on the presence of specific waste objects. In particular, the number of labeled training images is 547 while the number of labeled test images is 168.
- **Weakly-supervised localization**: a subset of 169 test images is annotated with segmentation masks surrounding relevant waste objects.

In this thesis, multi-class multi-label annotations are used to design a model that can distinguish among the different types of waste that are present in an illegal landfill. The data set takes into consideration 22 different categories, even though images may also contain other types of non-annotated items. The available categories can be grouped into two different macro-categories:

- **Waste types** (15): the classes that are part of this macro-category represent the type of objects that can be present in a landfill. The annotated classes are *Rubble/excavated earth and rocks, Bulky items, Fire Wood, Scrap, Plastic, Vehicles, Tires, Domestic appliances, Paper, Sludge-Zootechnical waste-Manure, Stone/marble processing waste, Asphalt milling, Corrugated sheets (presumed asbestos-cement), Glass, Foundry waste.*
- **Storage modes** (7): the classes that are part of this macro-category are related to the type of container or the modality used to store the waste. The annotated classes are *Heaps not delimited, Container, Big bags, Pallets, Delimited heaps (by barriers/walls/etc), Cisterns, Drums bins.*

Figure 3.1 and Figure 3.2 present examples of images' crops of different waste types

[2] https://developers.google.com/maps/documentation/maps-static/overview

and storage modes labels. Looking at these figures, it is possible to see that while some objects such as *Bulky items*, *Tires*, *Vehicles* and *Fire Wood* are easy to recognize, others such as *Scrap* and *Plastic* are quite difficult to find. In general, storage modes are easier to distinguish for the human eye. These figures do not display examples for *Domestic appliances*, *Corrugated sheets*, *Glass*, and *Drums bins* that are very difficult to identify in the data set.

Taking into consideration the available classes, Table 3.1 reports the number of fine-grained annotated images that contain each specific class out of the total 715 annotated ones. For instance, out of 715 annotated images, 294 images contain *Rubble/excavated earth and rocks* while only 8 of them contain *Glass.* Looking at this table, it is easy to see that the data set is highly imbalanced. Some classes like *Rubble/excavated earth and rocks, Bulky items* and *Heaps not delimited* contain a lot more samples than *Glass, Corrugated sheets* and *Asphalt milling.* This imbalance can make the problem of distinguishing between multiple types of landfills more difficult given that DL architectures require lots of data to properly learn.

Table 3.1: The table reports the total number of annotated images for each class, along with the number and percentage of annotated images per class in the training and test set. The last row reports the total number of annotations. This number counts the same image multiple times, according to the number of annotated classes it contains.

Label	**#Images**	**#Training images**	**%Training images**	**#Test images**	**%Testing images**
Rubble/excavated earth and rocks	294	228	77.55	66	22.45
Bulky items	286	242	84.62	44	15.38
Fire Wood	173	135	78.03	38	21.97
Scrap	167	140	83.83	27	16.17
Plastic	126	102	80.95	24	19.05
Vehicles	53	27	50.94	26	49.06
Tires	45	32	71.11	13	28.89
Domestic appliances	24	19	79.17	5	20.83
Paper	26	21	80.77	5	19.23
Sludge-Zootechnical waste-Manure	19	15	78.95	4	21.05
Foundry waste	9	8	88.89	1	11.11
Stone/marble processing waste	13	12	92.31	1	7.69
Asphalt milling	12	9	75.00	3	25.00
Corrugated sheets	11	10	90.91	1	9.09
Glass	8	6	75.00	2	25.00
Heaps not delimited	448	355	79.24	93	20.76
Full container	167	113	67.66	54	32.34
Big bags	50	31	62.00	19	38.00
Full pallets	50	43	86.00	7	14.00
Delimited heaps	69	38	55.07	31	44.93
Cisterns	35	26	74.29	9	25.71
Drums bins	18	16	88.89	2	11.11
TOTAL	**2,103**	**1,628**	**77.33**	**475**	**22.67**

Table 3.2 reports the number of instances for each specific class for which a segmentation mask is provided. For instance, considering the 169 segmented test images, 131

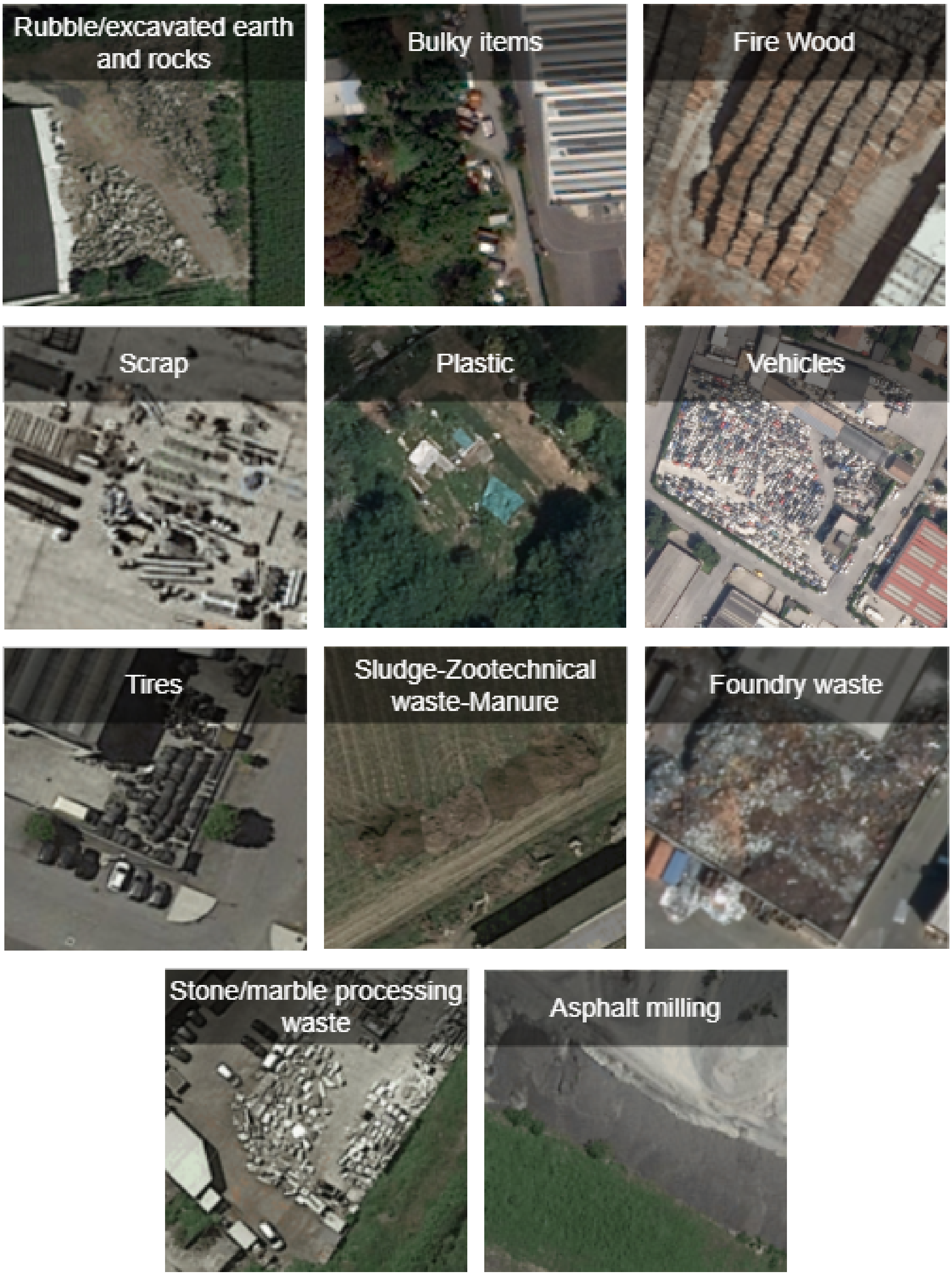

Rubble/excavated earth and rocks
Bulky items
Fire Wood
Scrap
Plastic
Vehicles
Tires
Sludge-Zootechnical waste-Manure
Foundry waste
Stone/marble processing waste
Asphalt milling

Figure 3.1: Examples of images annotated with different waste types labels. The used images are cropped from the AerialWaste data set [18].

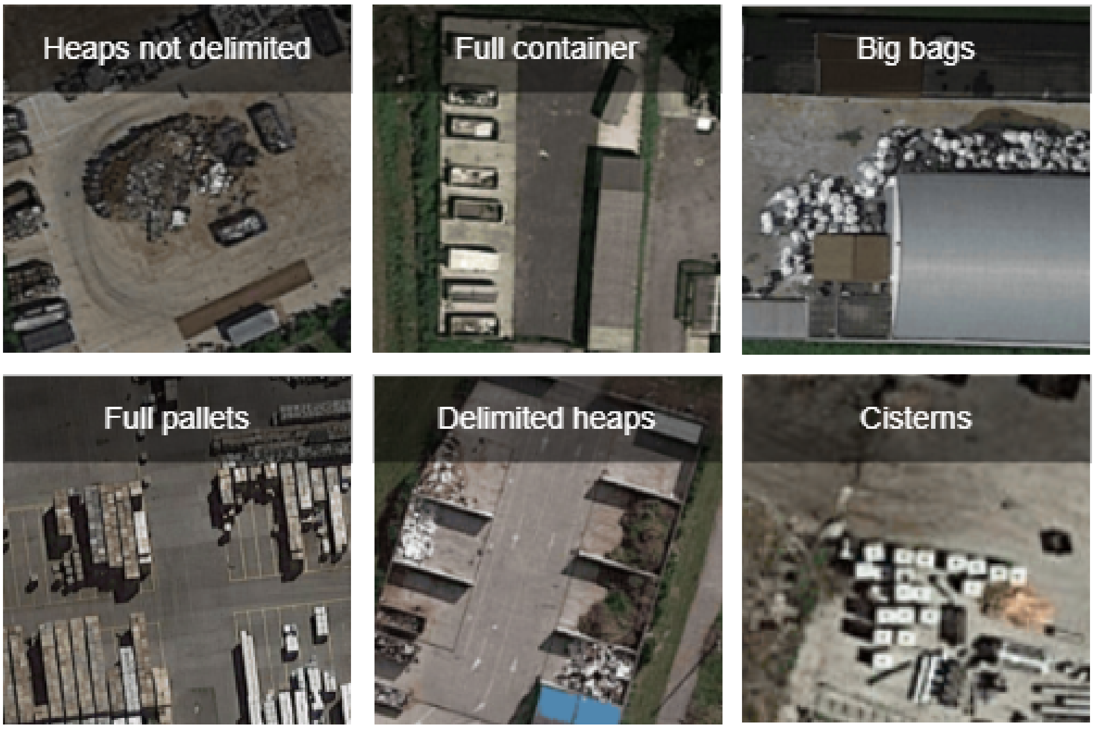


Figure 3.2: Examples of images annotated with different storage mode labels. The used images are cropped from the AerialWaste data set [18].

instances of *Rubble/excavated earth and rocks* are delineated by polygonal BBs. Looking at the table, it is evident that for many classes no BB is provided.

Table 3.2: The table reports the total number of segmented instances for each class. This number can be higher then the number of segmented images (169) given that each image may contain several instances of the same class.

Label	**#Segmented instances**
Rubble/excavated earth and rocks	131
Bulky items	69
Fire Wood	63
Scrap	32
Plastic	0
Vehicles	71
Tires	36
Domestic appliances	0
Paper	0
Sludge-Zootechnical waste-Manure	0
Foundry waste	0
Stone/marble processing waste	0
Asphalt milling	0
Corrugated sheets	0
Glass	0
Heaps not delimited	172
Full container	188
Big bags	60
Full pallets	0
Delimited heaps	55
Cisterns	0
Drums bins	0
TOTAL	**877**

As already outlined, the available classes are not always easily recognizable. This is caused by the fact that the images in the data set possess all the characteristics defined in Section 2.6.1 such as high intra-class diversity and high inter-class similarity due to the following factors:

- The area that is captured by the photograph can be an isolated site, an industrial area, or an urban area.
- The waste objects that are present in the images are extremely heterogeneous both in terms of scale and appearance.
- The spatial arrangement of the waste objects in the scene is diverse.
- Many different types of waste can be present at the same time in the same image, often close to each other or even partially overlapping.
- Some types of landfills are already intrinsically similar from a visual perspective (e.g., *Rubble* and *Manure*).

Figure 3.3 exemplifies these characteristics. Figure 3.4 shows a few examples of images from the training set confirming the previous considerations and highlighting that it is hard to distinguish among the different waste types. The complexity is further increased

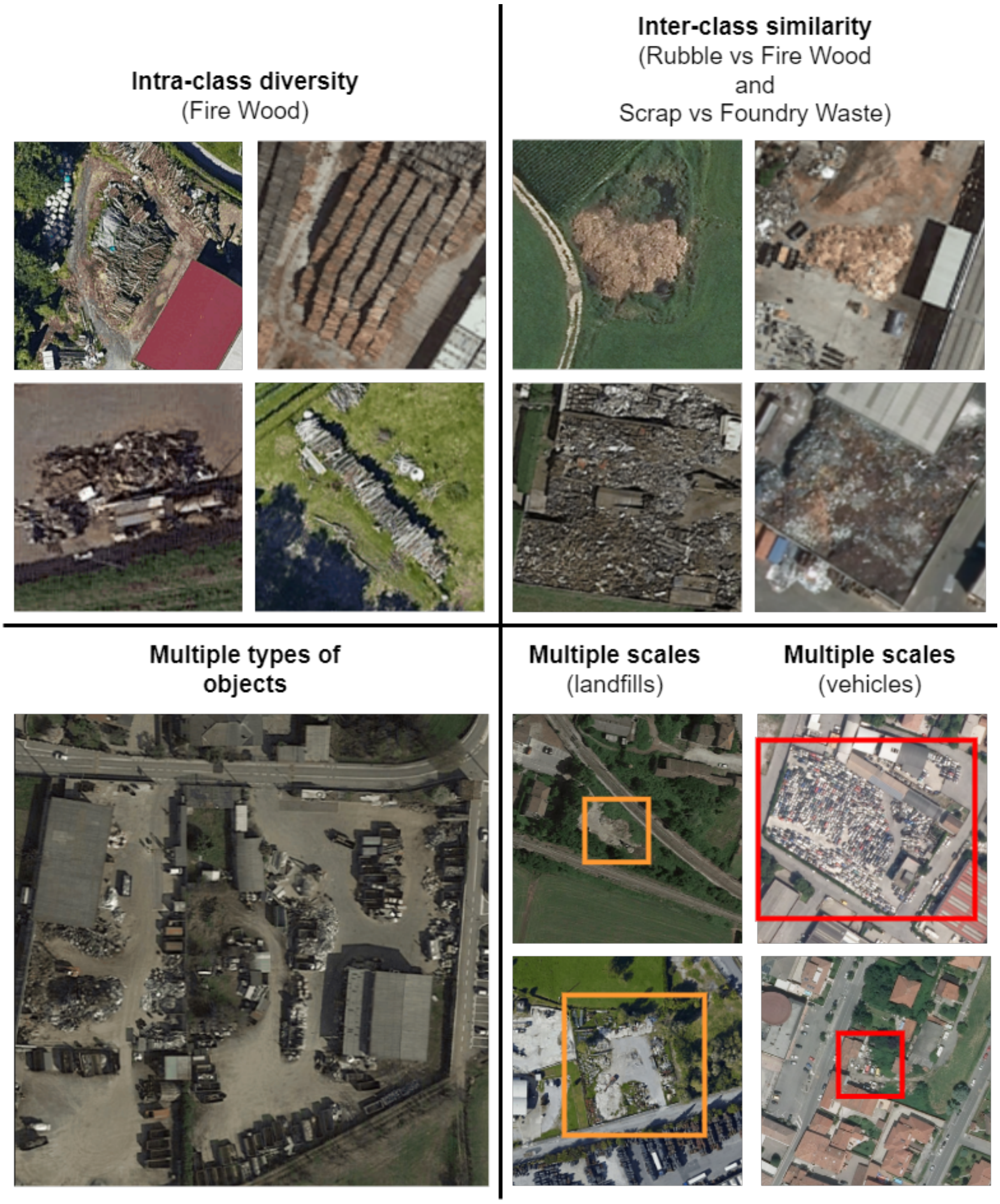
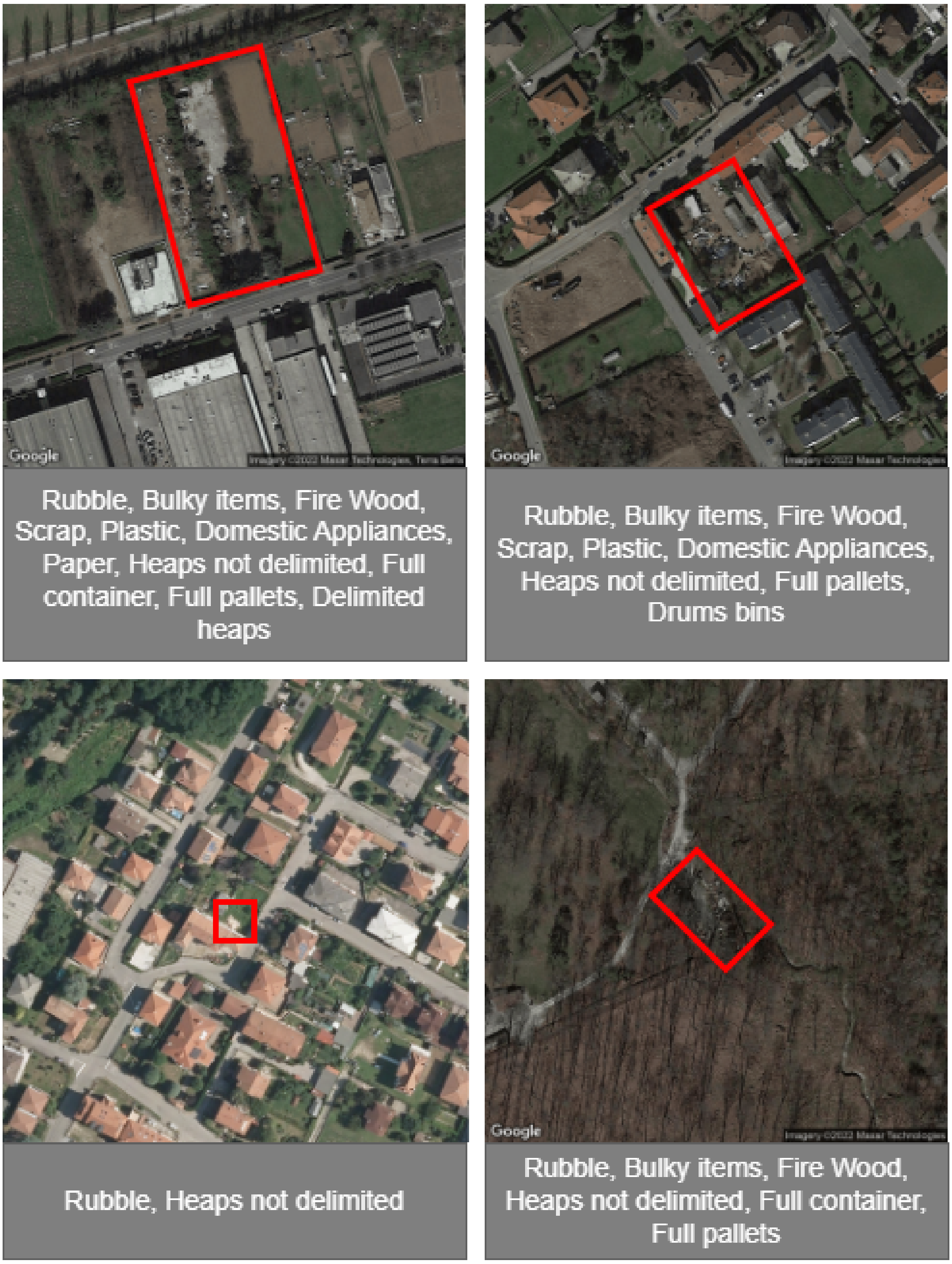


Figure 3.3: Landfills data set peculiarities. The used images are cropped from the AerialWaste data set [18].

by the large context. Figure 3.5 shows examples of test images with the corresponding segmentation masks, proving that the same considerations hold for the images present in the test set.

To gain more knowledge on the feasibility of differentiating among the different classes, it is possible to examine the co-occurrence matrix. The co-occurrence matrix gives an idea of how many times each class appears together with another. For instance, considering the *waste types*, Figure 3.6 reports the relative co-occurrence matrix: each row shows the number of times that a class appears with another class (absolute co-occurrence) divided by the total number of images in which the considered class appears. While the absolute co-occurrence matrix should be symmetric, the relative co-occurrence matrix is not symmetric. As it is possible to notice, especially for the classes *Rubble, Bulky items, Fire Wood, Scrap* and *Plastic*, the values in the co-occurrence matrix are quite high. For instance, considering the first row, it is possible to see that around 60% of the times in which *Rubble* appears, there are also *Bulky items* and around 30% of the times *Fire Wood.* Values on the right part of the matrix are much smaller, but this is due to the limited number of samples of these classes. Of course, high values in the co-occurrence matrix tend to jeopardize the discriminative power of the classifier. In fact, if two classes appear most of the time together and very few times alone, the risk is that the network tends to predict the presence or absence of both classes without understanding their differences, meaning without learning discriminative features that allow distinguishing among the two. Unfortunately, this is often the case for the considered data set.

The above considerations clearly show that understanding whether there is a disposal waste site or not, starting from RGB RSIs can be challenging and that being able to differentiate among the different items that are part of the image can be more complicated. The imbalance of the data set, the limited number of samples, and the high co-occurrence make this task even more complex. In this study, different options were considered and analyzed in Section 3.3 to address these complications.



Rubble, Bulky items, Fire Wood, Scrap, Plastic, Domestic Appliances, Paper, Heaps not delimited, Full container, Full pallets, Delimited heaps

Rubble, Bulky items, Fire Wood, Scrap, Plastic, Domestic Appliances, Heaps not delimited, Full pallets, Drums bins

Rubble, Heaps not delimited

Rubble, Bulky items, Fire Wood, Heaps not delimited, Full container, Full pallets

Figure 3.4: Examples of training images from the AerialWaste data set [18]. The red BBs give an approximate indication of where the landfill could be.

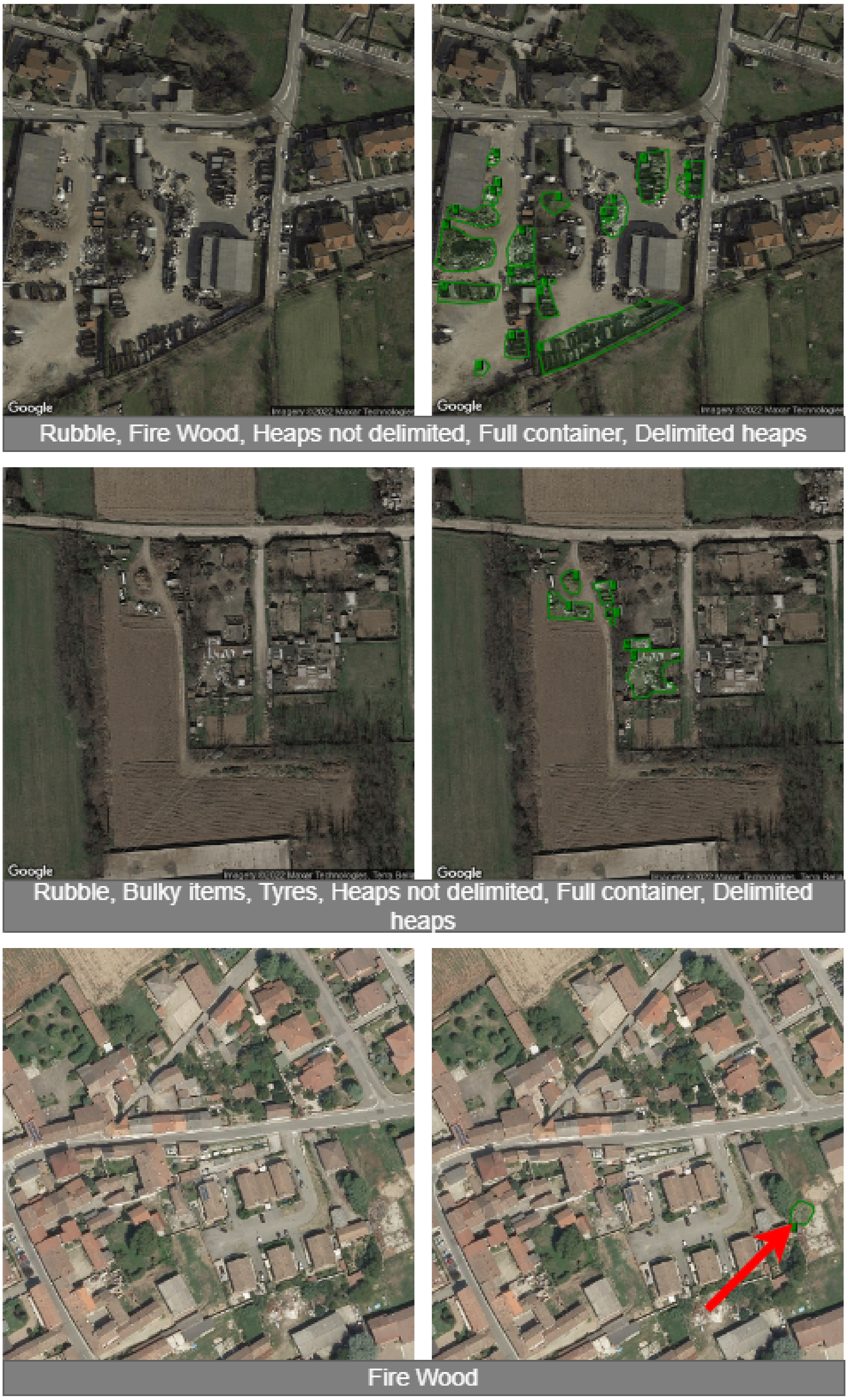
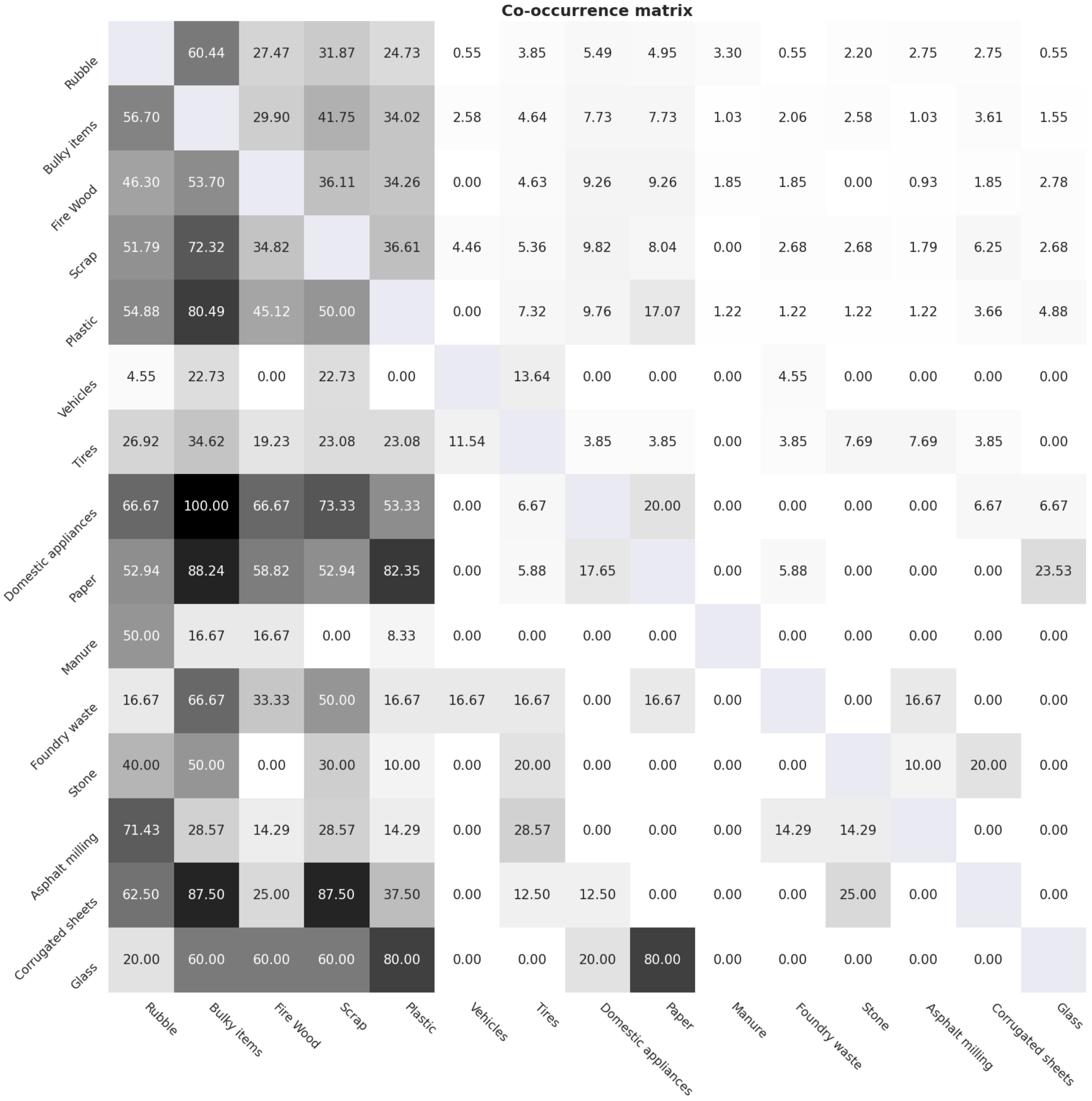

Google
Google
Rubble, Fire Wood, Heaps not delimited, Full container, Delimited heaps
Google
Google
Rubble, Bulky items, Tyres, Heaps not delimited, Full container, Delimited heaps
Fire Wood

Figure 3.5: Examples of testing images from the AerialWaste data set [18]. On the right, segmentation masks are shown. As it is possible to see, annotating the data set can be very complicated given that many objects are present in the same image, and sometimes, the objects of interest cover a minor portion of the image.



Co-occurrence matrix

	Rubble	Bulky items	Fire Wood	Scrap	Plastic	Vehicles	Tires	Domestic appliances	Paper	Manure	Foundry waste	Stone	Asphalt milling	Corrugated sheets	Glass
Rubble		60.44	27.47	31.87	24.73	0.55	3.85	5.49	4.95	3.30	0.55	2.20	2.75	2.75	0.55
Bulky items	56.70		29.90	41.75	34.02	2.58	4.64	7.73	7.73	1.03	2.06	2.58	1.03	3.61	1.55
Fire Wood	46.30	53.70		36.11	34.26	0.00	4.63	9.26	9.26	1.85	1.85	0.00	0.93	1.85	2.78
Scrap	51.79	72.32	34.82		36.61	4.46	5.36	9.82	8.04	0.00	2.68	2.68	1.79	6.25	2.68
Plastic	54.88	80.49	45.12	50.00		0.00	7.32	9.76	17.07	1.22	1.22	1.22	1.22	3.66	4.88
Vehicles	4.55	22.73	0.00	22.73	0.00		13.64	0.00	0.00	0.00	4.55	0.00	0.00	0.00	0.00
Tires	26.92	34.62	19.23	23.08	23.08	11.54		3.85	3.85	0.00	3.85	7.69	7.69	3.85	0.00
Domestic appliances	66.67	100.00	66.67	73.33	53.33	0.00	6.67		20.00	0.00	0.00	0.00	0.00	6.67	6.67
Paper	52.94	88.24	58.82	52.94	82.35	0.00	5.88	17.65		0.00	5.88	0.00	0.00	0.00	23.53
Manure	50.00	16.67	16.67	0.00	8.33	0.00	0.00	0.00	0.00		0.00	0.00	0.00	0.00	0.00
Foundry waste	16.67	66.67	33.33	50.00	16.67	16.67	16.67	0.00	16.67	0.00		0.00	16.67	0.00	0.00
Stone	40.00	50.00	0.00	30.00	10.00	0.00	20.00	0.00	0.00	0.00	0.00		10.00	20.00	0.00
Asphalt milling	71.43	28.57	14.29	28.57	14.29	0.00	28.57	0.00	0.00	0.00	14.29	14.29		0.00	0.00
Corrugated sheets	62.50	87.50	25.00	87.50	37.50	0.00	12.50	12.50	0.00	0.00	0.00	25.00	0.00		0.00
Glass	20.00	60.00	60.00	60.00	80.00	0.00	0.00	20.00	80.00	0.00	0.00	0.00	0.00	0.00	

Figure 3.6: Co-occurrence matrix of the waste types classes of the data set. Each cell represents the percentage of images in which the category on the row appears together with the category on the column. This co-occurrence matrix is generated using ODIN [19, 20].

3.2. Architecture

The authors of the AerialWaste data set [8, 18] propose a novel architecture to address the task of binary illegal landfills classification. More specifically, the classifier exploits a residual network [21] (ResNet50) as a backbone network augmented with a Feature Pyramid Network (FPN) [22]. This architecture is used to perform classification and then to generate CAMs for a better understanding of the localization and discriminative capabilities acquired by the network. Since the problem addressed in this thesis is strictly related to the work by Torres et al. [8, 18], their architecture is considered as a baseline for the experiments presented in Chapter 4.

3.2.1. Residual Network

The concept of residual network was first introduced by He et al. in 2015 [21]. The way in which CNNs are built allows to stack many many layers to solve more complex tasks. The deeper the network, the richer the learned features, thus increasing the number of layers of a network should improve the performance on complex tasks. However, as observed in [21], at a certain point, the performance tends to saturate and then slowly degrade. The study shows that this behavior is not due to overfitting since it is also observed during training and not only at test time. A more probable cause is the problem of vanishing or exploding gradient [283]. To gain more knowledge on this phenomenon and to solve it, the authors propose the introduction of skip connections. Skip connections alleviate the issue of vanishing gradient by introducing an alternate shortcut for the gradient to pass through, building a so-called *residual block*, shown in Figure 3.7. Residual blocks can be built with an arbitrary number of convolutional layers whose feature maps are eventually merged with the skip connection (identity branch). The insertion of skip connections enables the model to learn an identity function, ensuring that the higher layers of the model do not perform any worse than the lower layers. As a result, residual networks make it possible to train much deeper networks.

In this study, the adopted backbone is ResNet50 which consists of 5 stages, 4 of which are composed of residual blocks and whose details are reported in Table 3.3.

3.2.2. Feature Pyramid Network

Even though a residual network is already able to provide impressive performance in many cases, Torres et al. [8, 18] propose to augment it with an FPN to deal with the presence of multi-scale objects in the considered data set. An FPN [22] is a top-down

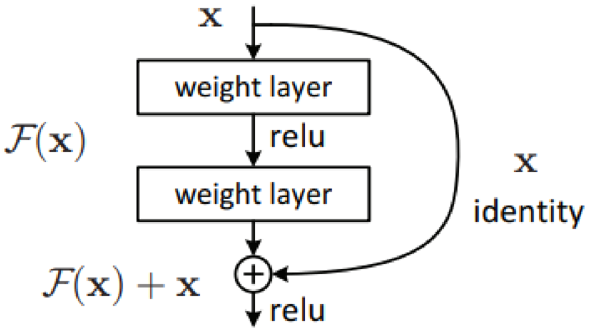

x
weight layer
$\mathcal{F}(\mathbf{x})$
relu
weight layer
x
identity
$\mathcal{F}(\mathbf{x}) + \mathbf{x}$
relu

Figure 3.7: Residual block. The gradient is allowed to backpropagate through two alternative paths, thanks to a skip connection, mitigating the problem of the vanishing gradient. Image taken from [21].

architecture with lateral connections developed for building high-level semantic feature maps at all scales. The traditional architecture of an FPN is shown in Figure 3.8.

Two different paths are present: a bottom-up and a top-down pathway. The bottom-up pathway is the feed-forward computation of the backbone network. Each stage (set of layers) of the backbone network corresponds to a layer of the pyramid. Then, the output of each stage is kept apart to maintain additional sets of feature maps for enriching the top-down pathway. Once the feed-forward computation has been performed, to merge the information gained by the different layers of the pyramid, it is necessary to upsample feature maps. More precisely, deeper feature maps usually capture semantically stronger elements and possess lower resolution than shallower layers. Thus, it is necessary to perform an upsampling of deeper feature maps to match the resolution of shallower ones before merging the information. The top-down pathway is in charge of upsampling deeper feature maps and then merging them with the shallower ones. The merging operation is performed by exploiting the lateral connection which can be a simple element-wise addition or a more complex set of operations. Each lateral connection merges feature maps of the same spatial size from the bottom-up pathway and the top-down pathway. The feature maps from the bottom-up pathway are passed through a *bottleneck layer* which consists of a 1×1 convolution to reduce the channel dimension. By exploiting an FPN, it is possible to learn features at different scales which can boost the performance in case multi-scale objects are present [284, 285].

The overall architecture proposed by Torres et al. [8, 18] is shown in Figure 3.9. In the Figure, C_2, C_3, C_4, C_5 denote the feature maps (outputs) produced by the corresponding stages of the ResNet50: $conv2, conv3, conv4, conv5$ (Table 3.3). A batch of B

Table 3.3: ResNet50 architecture details. The output size is calculated considering as input an image of size $H \times W$. The network has around 23 million trainable parameters.

Stage name	**Output size**	**ResNet50**
1 (conv1)	$\frac{H}{2} \times \frac{W}{2}$	$\begin{bmatrix} 7 \times 7, 64,\ stride\ 2 \\ 3 \times 3\ max\ pool,\ stride\ 2 \end{bmatrix} \times 1$
2 (conv2)	$\frac{H}{4} \times \frac{W}{4}$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
3 (conv3)	$\frac{H}{8} \times \frac{W}{8}$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$
4 (conv4)	$\frac{H}{16} \times \frac{W}{16}$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$
5 (conv5)	$\frac{H}{32} \times \frac{W}{32}$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$
	1×1	average pool, 1000-d fc, softmax

RGB images, each with height H and width W, is passed through the bottom-up path. The output of the backbone bottom-up path is a volume of size $[B, H/32, W/32, 2048]$. Then, the top-down path follows the previously explained idea of an FPN. More specifically, each feature map C_i of size $[B, H/2^{i+1}, W/2^{i+1}, 2^{7+i}]$ is passed through a bottleneck layer to reduce the number of channels to 256, producing a new feature map B_i of size $[B, H/2^{i+1}, W/2^{i+1}, 256]$. Then, a merging operation is performed to obtain the feature map M_i. More specifically, the merging operation is performed through a lateral connection that consists of a concatenation in the channel dimension, while the upsampling procedure needed to increase the resolution of deeper feature maps is done through bilinear interpolation. The resulting upsampled merged feature map M_i can be computed by the following expression:

$$M_i = \begin{cases} B_5, & i = 5 \\ Concat(\ B_i\ ,\ Bilinear2x(\ M_{i-1}\ )\ ), & otherwise \end{cases}$$

Due to the concatenation, the upsampled merged feature map's channel dimension increases while moving from top to bottom. Specifically, the size is given by $[B, H/2^i, W/2^i, 256 \times (6 - i)]$. The merged features are de-aliased to obtain the pyramid layers P_2, P_3, P_4, P_5. Then, each P_i is subjected to Global Average Pooling (GAP) followed by a flattening op-

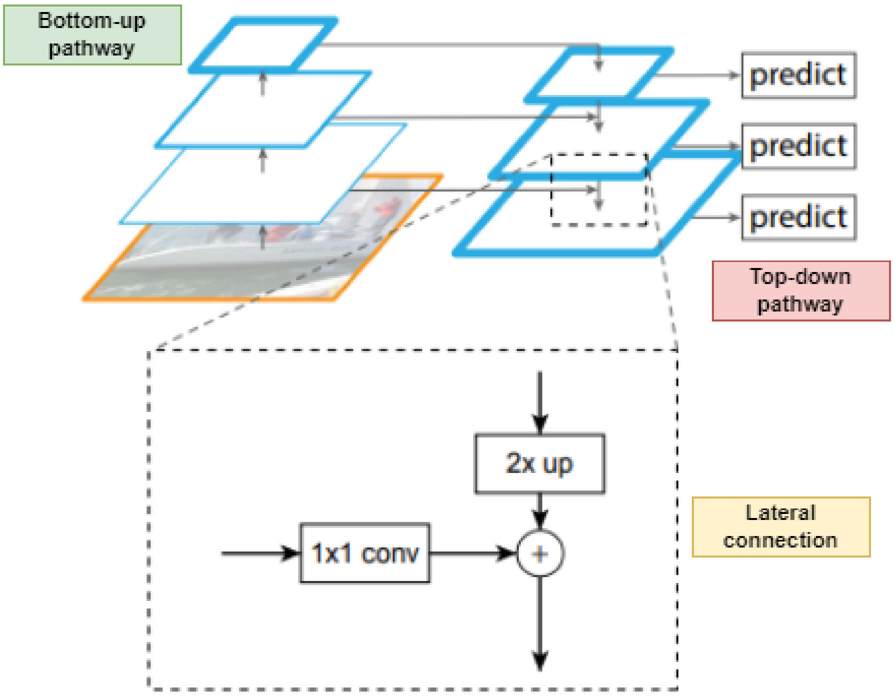


Figure 3.8: Typical architecture of an FPN. Image modified from [22].

eration to produce the vectors P_2', P_3', P_4', P_5'. Each P_i' is passed through a Fully Connected (FC) layer to perform classification at the respective scale level. Finally, the predictions of each level are concatenated and fed to the final FC layer to produce the final prediction. The final FC layer can have two or more outputs depending on whether binary or multi-label classification is performed. In both cases, a sigmoid activation function is applied to the final prediction vector to output a probability value indicating whether each class is present or not.

3.2.3. Class Activation Map

Given that DL models are difficult to interpret, it is possible to exploit explainability methods to better understand how the network is producing its predictions. More specifically, CAMs [225] are a simple but very useful mechanism to highlight the portions of the image which have a higher influence on the output of the model.

Specifically, a CAM is a weighted activation map that can be generated for each sample at inference time. CAMs can be obtained without the need for any specific supervision, making them a milestone for many weakly supervised detection approaches, as described in Chapter 2. To generate CAMs, the architecture needs a Global Average Pooling (GAP) layer between the last convolutional layer and the classifier. The addition of the GAP

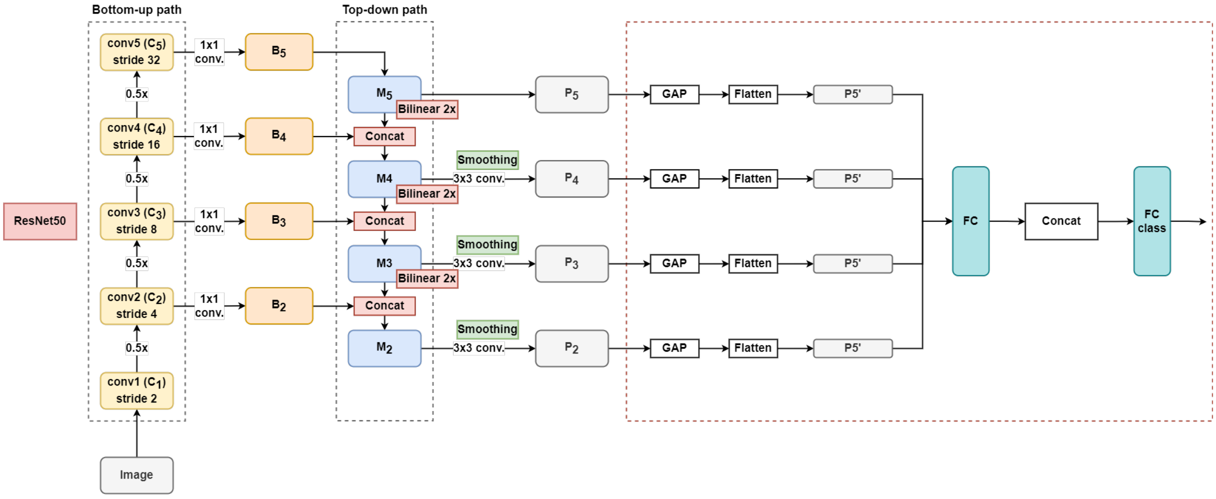


Figure 3.9: Employed architecture. Image modified from [8].

layer allows to compute the average of each feature map and to sum up the spatial information, acting also as a structural regularizer and generating outputs more robust to spatial translations of the input.

Given an image, the CAMs (one for each class) can be obtained by a weighted sum of the feature maps of the last convolutional layer. The weights are obtained from the classifier. The final CAMs usually have a smaller size with respect to the original input, thus they are upsampled (e.g., with bilinear interpolation) to match the input dimension.

Since part of the aim of this work is to localize the landfills inside of the image, the generation and analysis of the CAMs are particularly useful to assess the comprehension of the different types of landfills and their position during the classification task. A qualitative evaluation is even more relevant for the considered task given the characteristics and issues related to the employed data set (Section 3.1), which can affect the classification results. An in-depth analysis of the CAMs can complement the quantitative evaluation of the classification results and shed light on the actual level of comprehension reached by the network. With this aim, two different types of CAMs are taken into consideration in this study:

- **Intermediate CAMs**: it is possible to produce a CAM taking into consideration only the output of a single layer of the pyramid. This results in the generation of a CAM for each layer and each class, indicating the portions of the image on which each layer focuses more. Given that the considered architecture has four different pyramid layers, four intermediate sets of CAMs are produced. More specifically, the feature maps produced by each pyramid layer (P_2, P_3, P_4, P_5) are multiplied by the learned weights of the FC layer (the one before the concatenation). Then, the output of the product of P_i by the weights of the FC layer is multiplied by the weights of the final FC layer (FC class) related to the i^{th} pyramid layer of the network. In this way, an intermediate CAM is obtained for each category and each pyramid layer.

- **Global CAM**: once the intermediate CAMs have been obtained, it is possible to merge them into a final set of CAMs to obtain a global CAM for each class, that captures the important aspects of the intermediate ones.

Given the presence of multi-scale objects, it is possible to generate a CAM for different image scales and eventually fuse them in a single CAM which can be more precise and more robust to the different object scales. To obtain the multi-scale global CAM of an image I of size $[H, W]$, it is necessary to rescale the image according to different scale ratios. In this study, each image is rescaled according to the ratios $R = [0.5, 1.0, 1.5]$. In this way, three images are obtained: one for each scale ratio r, whose size is $[r \times H, r \times W]$. Each rescaled image I_r is passed through the network, and a CAM is computed. These CAMs are then upsampled to the same dimension. Finally, the multi-scale global CAM can be obtained by performing an element-wise sum of the CAMs obtained for every I_r. A CAM is always obtained for each category.

3.2.4. Loss Function

The illegal landfills classification problem considered in this study belongs to the family of multi-label classification problems. In this scenario, the Ground Truth (GT) label of a sample n is represented as a binary vector, say Y_n, of size C, where C is the number of classes. Each element $y_n^i \in \{0, 1\}$ of vector Y_n, indicates the presence ($y_n^i = 1$) or absence ($y_n^i = 0$) of class i for the considered sample. For instance, considering the set of classes $\mathbf{C} = \{\textit{Bulky items, Scrap, Vehicles}\}$, an image with GT label $Y_0 = [1, 0, 1]$ contains at least one instance of *Bulky Items* and at least one instance of *Vehicles*, while no instance of *Scrap* is present in the image.

The output of the network is a prediction represented by a binary vector, X_n of the same size as Y_n. Each element $x_n^i \in [0, 1]$ of vector X_n represents the estimated likelihood of the presence of class i in the considered sample, computed by the network. For example, considering the same set of classes $\mathbf{C}$ as before, a prediction $X_0 = [0.30, 0.67, 0.95]$ means that the model estimates that the likelihood of the presence of at least an instance of *Bulky Items* is 0.3, that of at least an instance of *Scrap* is 0.67 and the likelihood of the presence of at least an instance of *Vehicles* is 0.95.

Multi-label Soft-Margin Loss

This scenario allows to recast the multi-label classification problem into multiple binary classification problems, where every element x_n^i represents the binary classification score for the class i and sample n and y_n^i is the corresponding binary GT label for the same class. Because of that, the loss function for multi-label classification can be considered an extension of the loss function that is used for binary classification, named *Binary Cross-Entropy* (BCE), given that the only difference between the two tasks is the number of classes to which a sample belongs. In the case of a single class, the vectors X_n and Y_n collapse to scalars x_n and y_n for every element n, and the multi-label loss function becomes equivalent to the BCE loss function. For each sample n in the data set, with prediction-label pair given by (x_n, y_n), the BCE loss can be computed as reported in Equation 3.1.

$$BCE(x_n, y_n) = -y_n \cdot \log \sigma\left(x_n\right) + \left(1 - y_n\right) \cdot \log\left(1 - \sigma\left(x_n\right)\right) \tag{3.1}$$

In the above equations, σ represents the sigmoid activation function given by:

$$\sigma(x_n) = \frac{1}{1 + e^{-x_n}}$$

If a sample belongs to more than one class, an extension of BCE loss called *Multi-label Soft-Margin Loss* can be used. Considering a single sample n, the prediction of the network X_n and the corresponding GT vector Y_n, the loss can be computed by averaging the BCE losses of the single classes $(BCE(x_n^i, y_n^i))$. Thus, for each sample n in the data set, with prediction-label pair given by (X_n, Y_n), the Multi-label Soft-Margin loss, is given by Equation 3.2.

$$\begin{aligned} \ell(X_n, Y_n) &= \frac{1}{C} * \sum_{i=1}^{C} BCE(x_n^i, y_n^i) = \\ &= -\frac{1}{C} * \sum_{i=1}^{C} \left[y_n^i * \log \sigma\left(x_n^i\right) + \left(1 - y_n^i\right) * \log\left(1 - \sigma\left(x_n^i\right)\right)\right] \end{aligned} \tag{3.2}$$

Finally, given that samples are passed through the network in batches, the batch loss can be computed by averaging the losses of all the samples in the batch. Considering a batch of N samples, where the output of the network is $X_N = \left[X_1, X_2, \ldots, X_n, \ldots, X_{|N|}\right]$ and the GT labels are $Y_N = \left[Y_1, Y_2, \ldots, Y_n, \ldots Y_{|N|}\right]$, the batch loss is described by Equation 3.3.

$$\begin{aligned}\mathcal{L}(X_N, Y_Y) &= \frac{1}{N} * \sum_{n=1}^{N} \ell(X_n, Y_n) = \\ &= -\frac{1}{NC} * \sum_{n=1}^{N} \sum_{i=1}^{C} \left[y_n^i * \log \sigma\left(x_n^i\right) + (1 - y_n^i) * \log\left(1 - \sigma\left(x_n^i\right)\right)\right]\end{aligned} \tag{3.3}$$

3.3. Proposed Approaches and Methods

Given the characteristics of the considered data set, especially concerning the imbalance nature and the high class co-occurrence, it is necessary to develop strategies that allow dealing, at least partially, with these issues. The different possible solutions that are analyzed and tested are described in this section. As previously stated, all these methods are built to improve the task of multi-class multi-label classification using the architecture described in Section 3.2, paying particular attention to improving the discriminative power and the localization capabilities of the network. For this reason, a couple of methods to generate segmentation masks starting from the results obtained from the classification model are analyzed. These segmentation masks can then be used as pseudo-labels for a weakly supervised localization task.

Offline Data Augmentation

Data sets with a limited number of samples are very frequent in modern applications. However, the data-hungry nature of DL architectures requires a high number of samples to properly learn from scratch. Of course, the trivial solution to the limited availability of data is to collect more of them. Unfortunately, this is not always feasible and inexpensive.

By analyzing the AerialWaste data set, it is evident that most images have a resolution of around 1,000x1,000 pixels, and the suspicious sites are often located in the center of the image. These characteristics allow to develop an efficient way of enlarging the data set by simply cropping each image in different patches of lower resolution. This operation can be performed offline (before training) and can be considered a data augmentation technique since it allows to augment the number and variability of the images fed to the DL architecture. In this study, each image is cropped in 5 different patches: a center patch, and a patch for each of the corners (top-left, top-right, bottom-left, bottom-right). The idea behind this choice is that if the illegal landfill is located in the middle of the image and the crop size is sufficiently big, in the majority of cases all the crops, or at least most of them, will contain a portion of the landfill. The major issue of this approach, especially in the case of multi-label classification is that mislabeled images may be generated. If multiple types of landfills are present in the same image, each patch is annotated with the same label as the original image. However, there is no guarantee that all crops contain all the classes of the original image. These mislabeled images can potentially result in noise being introduced during the training phase. The bigger the crop size, the higher the overlap between the patches but the lower the risk of wrongly annotating images. Nonetheless, a bigger crop size results also in a major influence of the image context. For this reason, it is necessary to achieve a trade-off between the context influence and the number of mislabeled samples by tuning the crop size.

Figure 3.10 shows, on the left, the idea of offline data augmentation.

Oversampling

Another frequent problem of data sets is related to class imbalance. This problem arises from the fact that the number of available samples for each class can be very different. To deal with this issue, several different techniques have been proposed over the years (e.g., undersampling, cost-sensitive learning [286]). Among them, a very intuitive approach is oversampling.

Oversampling consists in increasing the number of samples for the classes that are less represented. Once again, a trivial solution is to collect new data for some classes. Given that this is not always possible, a viable idea is to create multiple copies of the available images to augment the data set. A major risk of this approach is that if the network is fed with the exact same copy of an image multiple times, it can overfit. In this study, online data augmentation is performed to avoid this problem. More specifically, before being fed to the network (during training), each image can be randomly flipped (horizontally and vertically) and rotated by multiples of 90°. Thus, the network will rarely see the exact same image often, given the high number of possible online augmentations that are applied. Using oversampling, the number of available samples of specific classes can be increased, mitigating the problem of limited availability of samples. Moreover, by oversampling images containing more than one class, it is possible to alter the co-occurrence of the classes.

An example of how oversampling works is displayed in the center part of Figure 3.10.

Synthetic Data Augmentation

Oversampling can allow to partially deal with the high class co-occurrence and class imbalance issues. However, the benefits that it can potentially bring are strictly related to the number of samples that are already available. If a class has very few samples, the number of possible combinations of flip and rotate augmentations are still limited and may not be sufficient to guarantee that the network does not overfit. To solve this problem, it is possible to generate synthetic data.

Given the huge amount of negative samples in the AerialWaste data set, it is possible to generate synthetic data by inserting patches of illegal landfills on images without landfills. These synthetic images can then be used together with the real ones during training, increasing the number of samples and reducing the class co-occurrence and imbalance, depending on the strategy adopted to generate new samples, as in the case of oversampling. To implement this approach, images from the training set were analyzed and around 70 to 100 landfill patches were extracted for each considered class. Then, before inserting the patch on negative images, random flipping (vertical and horizontal) and rotation (between 0° and 360°) were applied to reduce the possibility of overfitting.

The main limitation of this approach is related to the realism of the generated images. A trivial way of generating a synthetic sample is to randomly select a background image and a landfill patch of a known type and then insert the patch in a random location on the background. In this way, a new image is obtained and labeled according to the type of landfill that is used. In this case, the generated images can be far from realistic, especially in case more than one patch is placed on the background or multiple types of landfills are added to the original image. To mitigate this issue, a first rough idea is to blur the contours of the patch before placing it on the background, reducing the contrast between the two images.

In many cases, however, blurring the contours is not sufficient to ensure the generation of realistic images since, often, negative samples are images of huge open fields where it is highly unrealistic to find an illegal landfill. Furthermore, even if an urban or industrial area is selected in place of a field as a negative sample, a random placement of the patch causes the generation of images in which landfills can appear in highly unrealistic spots such as roofs. To mitigate this issue, it is possible to use a CAM-guided approach. The idea is to exploit a model such as that of Torres et al. [8, 18] to make inference on the negative instances (those to be used as background). Even though most of the time, the network will hopefully correctly classify the samples as negative, the CAM can still provide an indication of where the network is focusing its attention. This can be considered a coarse suggestion of where a landfill could be placed. Thus, positioning a patch where the CAM has the highest value can sometimes be more realistic. Countermeasures are taken to avoid placing more than one patch in the same position, thus selecting different candidate locations. Using this approach, it is possible to obtain more realistic images and sometimes the result is impressive (right portion of Figure 3.10). However, the realism of CAM-guided generated images is still limited and dependent both on the dimension and appearance of the patch and especially on the capability of the model chosen to generate the CAMs.

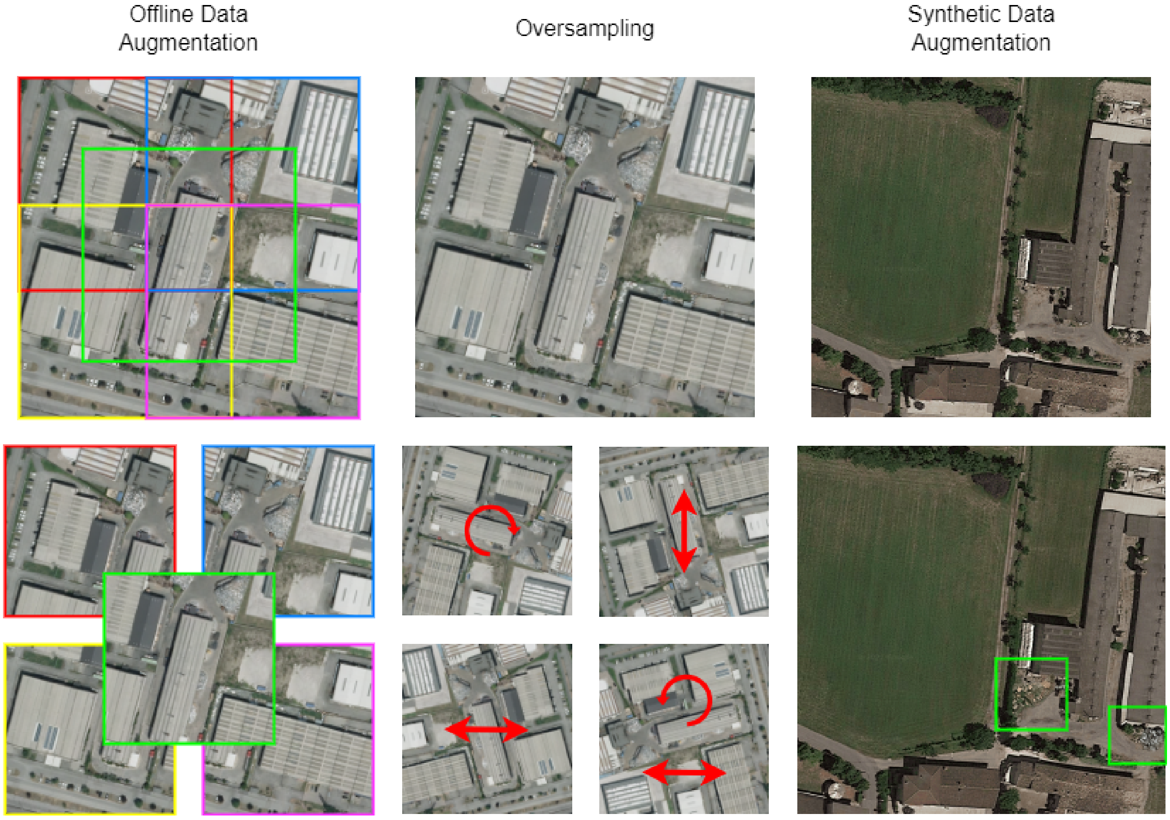


Figure 3.10: Examples of offline data augmentation, oversampling and CAM-guided synthetic data augmentation (SDA). At the top row, the image before the application of the indicated method is displayed. The bottom part displays the image(s) generated after the application of the different techniques.

3.3.1. Data Generation Strategies

As already explained, to mitigate the issues related to the characteristics of the data set, it is possible to augment the training set with new images. On the one hand, cropping the original images can allow to increase the size of the data set, but at the same time, it does not solve other problems such as class co-occurrence and imbalance. On the other hand, oversampling and SDA can help solve the other problems. In these cases, it is necessary to design a specific image generation strategy that is able to act on specific aspects of the data set. For instance, to reduce the co-occurrence of two classes, it is possible to generate a certain number of images in which the two classes do not appear together.

Uniform Strategy

Given the characteristics of the data set and specifically the reduced number of samples in which a class appears alone, a possibility is to generate a high number of synthetic samples in which only patches of a single class are used or, alternatively, perform oversampling only of those samples that already contain a single class. This simple strategy can already help to reduce class co-occurrence. In this study, a uniform strategy was considered, meaning that the number of samples generated is the same for each class. Still, it is possible to consider other strategies that can reduce not only the class co-occurrence but also the imbalance.

Simplex-guided Strategy

Even though the uniform strategy can be easily extended to a non-uniform one, the addition of synthetic data risks in principle to reduce the capabilities of the classifier since the employed network may overfit the generated data. At the same time, the realism of these images is limited. Thus, the addition of unnatural samples should be done carefully, and the number of newly added samples should be as limited as possible. This issue is not only related to synthetic data but also to data generated via oversampling since the addition of too many samples can potentially lead to overfitting. To limit the number of new samples, it is possible to compute the minimum number and type of samples that should be added to reduce the class co-occurrence and the imbalance of the data set. In this study, the sample selection problem is formulated as a *Linear Programming* problem and approached using the *Simplex Algorithm* [287]. The output of the algorithm is the number of samples to add to the original data set along with the classes that should be present in each sample so that the obtained data set possesses some pre-defined characteristics such as a low class co-occurrence.

Before introducing the constraints that allow the Simplex Algorithm to compute the needed samples, it is necessary to introduce a set of notions and functions that simplify the problem formulation.

Given a data set containing C different classes, the set $\mathcal{C} = \{c_0, c_1, \ldots, c_i, \ldots, c_{C-1}\}$ represents the available classes. Considering C unique classes, the power set of C is the set $\mathcal{P} = \{p_0, p_1, \ldots, p_i, \ldots, p_{P-1}\}$ containing all the $P = 2^C$ possible subsets of classes.

For instance, considering the set $\mathcal{C} = \{$*Fire Wood, Scrap, Vehicles*$\}$, $\mathcal{P}$ is given by:

$$\begin{aligned}\mathcal{P} =&\{\varnothing, \{Fire\,Wood\}, \{Scrap\}, \{Vehicles\}, \{Fire\,Wood, Scrap\}, \{Fire\,Wood, Vehicles\},\\ &\{Scrap, Vehicles\}, \{Fire\,Wood, Scrap, Vehicles\}\}\end{aligned}$$

Furthermore, let's consider a function $bin(x, n)$ that returns the binary representation of number x using n digits, as reported in the example below.

$$\begin{aligned} bin(0,3) &= [0,0,0] \qquad & bin(1,3) &= [0,0,1]\\ bin(5,3) &= [1,0,1] \qquad & bin(5,5) &= [0,0,1,0,1]\end{aligned}$$

To simplify the notation, let's suppose, without loss of generality, that a class c_j is part of the subset p_i if and only if $bin(i, C)_j = 1$. For instance, considering the set $\mathcal{C} = \{$*Fire Wood, Scrap, Vehicles*$\}$ and the subset p_3, $bin(i, C) = bin(3, 3) = [0, 1, 1]$. Thus, p_3 will only contain the second (c_1) and the third (c_2) classes of set $\mathcal{C}$: $p_3 = \{$Scrap, Vehicles$\}$. In the same way, p_4 will only contain class *Fire Wood* given that $bin(4, 3) = [1, 0, 0]$.

At this point, let's call $class(i)$ the function that, given an index i of a subset p_i, returns the set of indexes j of all the classes c_j contained by p_i. For example, considering again the set $\mathcal{C} = \{$*Fire Wood, Scrap, Vehicles*$\}$:

$$\begin{aligned} class(0) &= \varnothing \qquad & class(4) &= \{0\}\\ class(3) &= \{1,2\} \qquad & class(7) &= \{0,1,2\}\end{aligned}$$

In fact, the classes contained in p_3 are *Scrap* (c_1) and *Vehicles* (c_2), whose indexes are 1 and 2, while the ones of p_7 are *Fire Wood* (c_0), *Scrap* (c_1) and *Vehicles* (c_2), whose indexes are respectively 0, 1 and 2.

Vice-versa, $comb(j)$ is the function that, given an index j of a class c_j, returns the set of indexes i such that c_j belongs to p_i. For instance, taking for example the set $\mathcal{C} = \{$*Fire Wood, Scrap, Vehicles*$\}$:

$$comb(0) = \{3,5,6,7\} \qquad comb(2) = \{1,3,5,7\}$$

In fact, class *Fire Wood* whose index is 0, is part of p_3, p_5, p_6 and p_7, whose indexes are 3, 5, 6 and 7. The same reasoning holds for the class *Vehicles* with index 2, which is part of p_1, p_3, p_5 and p_7, with indexes 1, 3, 5 and 7.

Besides the above sets and functions, the following variables are considered:

- The number of samples n_i in the desired data set containing at least class c_j, for each $c_j \in \mathcal{C}$. These variable are grouped in the vector $N_{1 \times C} = [n_0, n_1, \ldots, n_i, \ldots, n_{C-1}]$.
- The number of samples m_i in the desired data set for every possible subset of classes p_i. These variables are grouped in the vector $M_{1 \times P} = [m_0, m_1, \ldots, m_i, \ldots, m_{P-1}]$. Notice that, for the augmentation purposes of this task, the element m_0 is not particularly relevant since adding an arbitrary number of samples without any class does not have any effect on the balancing ratio of the classes or the co-occurrence matrix.
- The absolute co-occurrence $z_{i,j}$ between each pair of classes c_i and c_j in $\mathcal{C}$, as defined in Section 3.1. The co-occurrence is stored in the following matrix:

$$Z_{C \times C} = \begin{bmatrix} z_{0,0} & z_{0,1} & \cdots & z_{0,C-1} \\ z_{1,0} & z_{1,1} & \cdots & z_{1,C-1} \\ \vdots & \vdots & \ddots & \vdots \\ z_{C-1,0} & z_{C-1,1} & \cdots & z_{C-1,C-1} \end{bmatrix}$$

  Notice that the elements on the main diagonal are not particularly meaningful, since there is no definition of co-occurrence between a class and itself.

As defined at the beginning of this section, the objective is to obtain the desired level of class co-occurrence and imbalance with the addition of the minimum number of possible samples. Thus, the objective function is the minimization of the total amount of samples, as reported in Equation 3.4.

$$\min \sum_{i=0}^{P-1} m_i \tag{3.4}$$

To accomplish this task, it is possible to define a set of parameters used to regulate the tightness of the various constraints. The role of the parameters will become clearer after the definition of the constraints. The considered parameters are:

- The maximum allowed relative co-occurrence $k_{i,j} \in [0, 1]$ between each pair of classes c_i and c_j in $\mathcal{C}$, as defined in Section 3.1. These parameters are grouped in the matrix:

$$K_{C \times C} = \begin{bmatrix} k_{0,0} & k_{0,1} & \cdots & k_{0,C-1} \\ k_{1,0} & k_{1,1} & \cdots & k_{1,C-1} \\ \vdots & \vdots & \ddots & \vdots \\ k_{C-1,0} & k_{C-1,1} & \cdots & k_{C-1,C-1} \end{bmatrix}$$

$k_{i,j} = 0$ means that no sample with class c_i will be allowed to contain also class c_j, while $k_{i,j} = 1$ means that every sample with class c_i is potentially allowed to always appear together with class c_j. Notice that the elements on the main diagonal must be set to 1 to allow the algorithm to converge, as will be clarified in the constraints' definition.

- The number of samples r_i for each subset of classes p_i in $\mathcal{P}$ present in the original data set. These parameters are grouped in the vector $R_{1\times P} = [r_0, r_1, \ldots, r_i, \ldots, r_{P-1}]$.
- The percentage $u_i \in [0, 1]$ of original samples containing the subset of classes p_i in $\mathcal{P}$ (represented by r_i) to keep in the final data set. These parameters are grouped in the vector $U_{1\times P} = [u_0, u_1, \ldots, u_i, \ldots, u_{P-1}]$. $u_i = 0$ means that all the original samples with a subset of classes p_i can be discarded if this allows to satisfy all the constraints and minimize the objective function. $u_i = 1$ means that all the original samples with a subset of classes p_i must be part of the final data set. This constrains the number of samples m_i containing the subset of classes p_i to be at least equal to the number of original samples r_i: $m_i \geq r_i \quad \forall i \in [0, P-1]$. In general, it is advisable to set $u_i = 1 \quad \forall i \in [0, P-1]$, since this forces not to discard any of the original samples. However, these values could be slightly reduced to allow to satisfy the constraints without the need to add too many samples. Moving these values far from 1 will lead to a data set with many discarded samples since this may be more convenient than adding new ones to minimize the objective function.
- The maximum allowed relative difference $b_{i,j} \in [0, 1]$ between the number of samples with class c_i, given by variable n_i, and the number of samples with class c_j, given by the variable n_j. These parameters are grouped in a matrix

$$B_{C\times C} = \begin{bmatrix} b_{0,0} & b_{0,1} & \cdots & b_{0,C-1} \\ b_{1,0} & b_{1,1} & \cdots & b_{1,C-1} \\ \vdots & \vdots & \ddots & \vdots \\ b_{C-1,0} & b_{C-1,1} & \cdots & b_{C-1,C-1} \end{bmatrix}$$

$b_{i,j} = 1$ means that n_i cannot be lower than n_j, while $b_{i,j} = 0$ means that n_i is not influenced by n_j. Notice that the elements on the main diagonal of B are trivially set to 1.

At this point, it is possible to define the constraints that need to be satisfied while minimizing the objective function. The first constraints to be defined are *bounding constraints*, needed to bound (relate) the values of the variables to each other. The variable

N can be bounded to M as defined in Equation 3.5. This stands from the fact that the number of samples of each class is equal to the sum of the number of samples for each subset of classes containing the considered one.

$$n_i = \sum_{j \,\in\, comb(i)} m_j \quad \forall i \in [0, C-1] \tag{3.5}$$

The bounding constraint between M and Z can be defined following the definition of co-occurrence given in Section 3.1, as reported in Equation 3.6. This stands from the fact that the co-occurrence of two classes is equal to the sum of the number of samples belonging to all the subsets of classes in which both the considered categories are present.

$$z_{i,j} = \sum_{w \,\in\, comb(i) \cap comb(j)} m_w \quad \forall i \in [0, C-1]\,, \quad \forall j \in [0, C-1]\,, \tag{3.6}$$

At this point, it is possible to define the constraints of the problem that, given the parameters, determine the final output given by the Simplex algorithm.

The *maximum co-occurrence constraint* forces the relative co-occurrence between classes c_i and c_j to be lower or equal to the maximum allowed relative co-occurrence of the two classes, as described in Equation 3.7. In this case, the constraint is formulated in terms of absolute co-occurrence given that this gives rise to a linear constraint and can be achieved by multiplying the maximum allowed relative co-occurrence of class c_i with other classes by the number of samples of class c_i.

$$z_{i,j} \leq k_{i,j} \times n_i \quad \forall i \in [0, C-1]\,, \quad \forall j \in [0, C-1] \tag{3.7}$$

This clarifies why the elements on the main diagonal of K must be equal to 1: given the bounding constraints described in Equations 3.5 and 3.6, it results that $z_{i,i} = n_i \quad \forall i \in [0, C-1]$, and, considering that the $k_{i,j} \in [0,1] \quad \forall i \in [0, C-1]\,, \forall j \in [0, C-1]$, the constraint described in Equation 3.7 is satisfied only for the values $k_{i,i} = 1 \quad \forall i \in [0, C-1]$.

Then, the *original sample usage constraint* forces to keep at least a certain amount of the original samples, imposing a lower bound for the values of M, as reported in Equation 3.8.

$$m_i \geq r_i \times u_i \quad \forall i \in [0, P-1] \tag{3.8}$$

The *balancing constraint* forces the number of samples per class, represented by N, to be as balanced as defined by matrix B, and it is described in Equation 3.9.

$$n_i \geq b_{i,j} \times n_j \quad \forall i \in [0, C-1], \forall j \in [0, C-1] \tag{3.9}$$

Now it is clear why the elements on the main diagonal of B are trivially set to 1: the constraint described in Equation 3.9, considering the case $i = j$, becomes $n_i \geq b_{i,i} \times n_i$ that is for sure satisfied if $b_{i,i} = 1 \quad \forall i \in [0, C-1]$.

At this point, the problem formulation can be extended to force other specific behaviors. In the case the augmentation strategy is oversampling, it is necessary to force the addition of elements that are already present in the original data set since there is no possibility to create copies of samples with the subset of classes p_i if $r_i = 0$. To do so, it is necessary to define the function $zeros(X)$ that, given an array $X_{1\times P} = [x_0, x_1, \ldots, x_i, \ldots, x_{P-1}]$, returns a set containing all the indexes i such that $x_i = 0$. For example, given the array $X = [0, 25, 0, 0, 43]$:

$$zeros(X) = \{0, 2, 3\}$$

A constraint can be defined to force this behavior, as reported in Equation 3.10.

$$m_i = 0 \quad \forall i \in zeros(R) \tag{3.10}$$

Another behavior that could be imposed is related to the fact that a higher number of classes in a sample could introduce more confusion since it is more difficult to learn the features of a specific class while there are more classes present in the samples. The behavior that can be imposed is that, for every class c_i, the number of samples with only the class c_i is forced to be greater or equal to the number of samples with class c_i and only another class, which in turn must be greater or equal than the number of samples with class c_i and other 2 classes, and so on. To obtain this behavior, let's introduce the function $super(i)$, that, given an index i of p_i in $\mathcal{P}$, returns the set of indexes j of p_j, such that $class(i) \subset class(j)$ and $|class(j)| = |class(i)| + 1$. For instance, considering the set $\mathcal{C} = \{$*Fire Wood, Scrap, Vehicles*$\}$, already cited in the previous examples:

$$super(1) = \{3, 5\} \qquad\qquad super(3) = \{7\}$$

In fact, $p_1 = \{$*Vehicles*$\}$, and the elements of $\mathcal{P}$ containing only *Vehicles* and just one other class are $p_3 = \{$*Scrap, Vehicles*$\}$ and $p_5 = \{$*Fire Wood, Vehicles*$\}$, whose indexes are 4 and 5. The same holds for $p_3 = \{\textit{Scrap, Vehicles}\}$, for which the only element of $\mathcal{P}$ containing *Scrap, Vehicles* and just one more class is $p_7 = \{\textit{Fire Wood, Scrap, Vehicles}\}$, whose index is 7.

A parameter $d_{1\times1}$ is introduced to indicate the minimum allowed difference, in percentage, between the elements with classes subset p_i, represented by m_i and the elements with classes subset $p_j \quad \forall j \in super(i)$, $\forall i \in [0, P-1]$, represented by m_j.

The final constraint is reported in Equation 3.11.

$$(1-d)\, m_i \geq \sum_{j \in super(i)} m_j \quad \forall i \in [0, P-1] \tag{3.11}$$

Once the Simplex algorithm converges to a solution, the variable M contains the number of samples for every class subset in $\mathcal{P}$ that needs to be present in the final data set. For every element $m_i \in M$, if $u_i = 1$, $m_i \geq r_i$, and $m_i - r_i$ represents the number of samples (synthetic or oversampled), with classes subset p_i, that need to be added to the original data set to obtain the final one. If $u_i < 1$, it is possible that $m_i - r_i < 0$, and, in this case, $r_i - m_i$ gives the number of samples with classes subset p_i that must be removed from the original data set.

3.3.2. Transfer Learning

To improve the possibility of obtaining more promising results given the complexity of the task, transfer learning is used to initialize the weights of the network. As already explained in Section 2.2.3, TL [59] is a solution for constructing data representations when the number of samples in the training set is limited as in the illegal landfills scenario. The basic idea is to transfer knowledge from a source task to a target one. The weights from a model trained on the source task are extracted and used as initialization for the target task model before eventually fine-tuning it.

Thus, instead of learning from scratch which is something particularly difficult, especially in the case of limited data and a complex task, the network's weights are initialized by exploiting the knowledge gained by solving a different task. More specifically, the following configurations are taken into account:

- **ImageNet pre-training**: the backbone network (ResNet50) is initialized using the weights obtained after performing classification on the wide-scale ImageNet [31] data set. This is a consolidated baseline for transfer learning which can boost performance in many cases. However, in this case, the source task (ImageNet classification) and the downstream one (Illegal Landfills multi-label fine-grained classification) are not very similar, especially from the domain viewpoint. This can potentially negatively impact the effectiveness of knowledge transfer.

- **Transfer learning from Torres et al. [18] model**: the backbone network is initialized using the weights obtained from the model provided by Torres et al. [18]. In this case, the source and target tasks are very similar, especially from the domain standpoint. Moreover, the knowledge obtained performing illegal landfills binary classification can be important to perform multi-label fine-grained classification of illicit waste disposal sites. If the network is not able to find illegal landfills, it is impossible to differentiate between them. Thus, this knowledge transfer may provide more benefits than ImageNet pre-training.
- **Self-Supervised Learning**: the backbone network is firstly trained on a pretext task based on self supervision. Once the task is completed, the weights of the backbone trained on the SSL task are transferred to the downstream task of Illegal Landfills classification. In this case, thanks to the nature of SSL, the wide amount of weakly labeled (labeled only at binary level) training samples can be exploited.

In all the configurations, the first two stages of the ResNet50 network are frozen before fine-tuning the downstream task. Moreover, the weights are loaded only for the backbone network. Section 3.3.3 describes the Self-Supervised approaches considered in this thesis.

3.3.3. Self-Supervised Approaches

Self supervision allows learning better feature representations. Over the years, several different successful approaches have been developed for both natural and RS images, as described in Section 2.3. Given the difficulty in discriminating between the different types of landfills and the availability of many weakly labeled samples, it is possible to exploit self-supervised approaches to verify whether the network is able to capture more meaningful and discriminative features for the different types of illegal landfills. In this study, three different approaches are considered and described: predicting image rotations, solving jigsaw puzzles, and Tile2Vec.

Predicting Image Rotations

Gidaris et al. [76] propose to learn better feature representations by predicting image rotations. More specifically, the idea is to rotate an image before feeding it to the network and try to predict its degree of rotation. Four possible rotations are considered (0°, 90°, 180°, 270°) meaning that the network outputs a distribution probability over four classes representing each specific rotation. Moreover, the four rotated images are fed to the network all at once since the authors report that this can improve performance. The underlying idea is that if the network can understand how an image is rotated, it has learned something about the content of the image itself. Of course, in the case of natural images, this task can be effective, while in the case of RSIs, it can be more complex to understand the image rotation given that images are usually taken from overhead. This task has already been used successfully by Zhao et al. [126] for RSIs scene classification.

Solving Jigsaw Puzzles

Noroozi et al. [99] propose to learn better feature representations by solving jigsaw puzzles. The idea is to extend the work presented in [79]. More specifically, a patch is first extracted from the image and divided into tiles. Each tile is represented by a number. In this way, the original patch can be represented by the sequence $S = 1, 2, 3, \ldots, N$ where N represents the number of tiles. The tiles are then mixed-up to obtain a puzzle represented by a vector S' of dimension N in which the numbers identifying the tiles are mixed-up in the same way as the tiles. The idea is to feed the mixed-up tiles to the network whose output should be S'. Thus, the network should learn how the tiles are mixed-up with respect to their original position. If the network can understand this task, it means that it can potentially understand the spatial relations of the image content. In the paper, the authors propose to apply a set of transformations (e.g., jittering) since they observed that if these transformations are not applied the network is able to learn shortcuts to solve the task without learning relevant features. As reported in [15] this approach is not often used for RSIs given that spatial correlation in overhead imagery is less dominant than in natural images. However, it is still considered in this study to evaluate its potential benefits for the illegal landfills' multi-label fine-grained classification task.

Tile2Vec

Recently, many researchers started to develop SSL techniques that can be directly applied to RSIs. The reason is that, as is often the case, approaches developed for natural images do not directly generalize to other more particular domains such as the RS one. A few years ago, Jean et al. [133] observe that geospatial analysis lacks pre-trained networks that significantly boost performance across a wide range of CV tasks. For this reason, the authors propose Tile2Vec, an unsupervised representation learning algorithm. In Natural Language Processing (NLP), Mikolov et al. [134] developed Word2Vec, a successful method to compute relevant word representations. The basic idea is that the distribution of words in a text is such that words appearing in similar contexts tend to have similar meanings. Tile2Vec tries to extend this idea to the spatial domain. However, it is necessary to define the right atomic unit (the equivalent of a word in NLP) and the right notion of context. The authors propose to consider a patch (a portion of an image) as an atomic unit and define the context based on the neighborhood of the patches. The assumption is that tiles (patches) that are close to each other have similar semantics and therefore they should on average have more similar representations than tiles that are far apart.

To implement this idea, the authors propose to follow three steps:

1. Select an anchor tile randomly from the original image.
2. Select a neighboring tile (positive) randomly in the neighborhood of the anchor tile.
3. Select a distant tile (negative) randomly far from the anchor tile.

The neighborhood of a tile is defined based on a parameter that represents the maximum distance at which a neighboring tile can be selected. The selected tiles represent a triplet. Each tile is passed through a CNN to extract features. Then, a *triplet loss* is applied to minimize the Euclidean distance between the anchor and neighboring tiles features while maximizing that between the anchor and distance tiles ones. The triplet loss is defined by Equation 3.12 in which (t_a, t_n, t_d) represent the features of the anchor, neighboring, and distant tiles respectively, while m is a term named margin whose purpose is to prevent the network from pushing the distant tile farther without restriction. Usually, the loss is regularized by introducing another term that penalizes the $L2 - norm$ of the representations to constrain the network to generate embeddings within a hypersphere, leading to a representation space in which relative distances have meaning.

$$\mathcal{L}\left(t_a, t_n, t_d\right) = \left[\left\|f_\theta\left(t_a\right) - f_\theta\left(t_n\right)\right\|_2 - \left\|f_\theta\left(t_a\right) - f_\theta\left(t_d\right)\right\|_2 + m\right]_+ \tag{3.12}$$

The most critical aspect of Tile2Vec is the choice of the tile size and the neighborhood, which can vary depending on the used data set. In this study, the possibility of improving the results of fine-grained multi-label classification is explored. In particular, the idea is to learn more useful features concerning the different types of illegal landfills. However, there is a major problem that needs to be faced before applying Tile2Vec to the AerialWaste data set: the tiles selection cannot be performed randomly. In fact, given that the landfills usually cover a small portion of the image, most of the time the selected tiles would not contain any waste disposal site. This would result in learning nothing about the different types of landfills which is instead fundamental for successful classification.

To solve this problem, a CAM-guided approach is proposed. The basic idea is based on the following two considerations:

- Models such as that of Torres et al. [18] are quite good at discovering whether there is an illegal landfill in an image.
- The CAMs obtained for the images can indicate where the illegal landfill (if present) is located.

Given the above considerations, the idea is to exploit the CAMs produced by an illegal landfill binary classification model to discover where an illicit waste disposal site may be located and then use this information to select tiles that are highly likely to contain a landfill.

To accomplish this task, the unlabeled images are first passed through the selected model to generate the CAMs and the classification scores indicating the likelihood of the presence of an illicit landfill. Once all the CAMs are produced, it is possible to start generating the tiles' triplets. The first step is to select an unlabeled image between those available. The choice takes into consideration only those images for which the classification score is greater than a predefined threshold. In this way, it is possible to avoid the selection of images that the model considers as negatives or for which it is not highly confident.

Once an image has been selected, the anchor, neighboring, and distant tiles are extracted in the following way (see Figure 3.11):

- **Anchor tile selection**: the anchor tile needs to be a small portion of the image that contains an illegal landfill. For this reason, the center point of the tile is selected among the points for which the CAM is over a certain threshold. The higher the threshold, the more likely the selected point is to indicate the position of an illegal landfill.
- **Neighboring tile selection**: the neighboring tile needs to be very close to the anchor tile. The reason is that if the neighborhood is big, either the neighboring tile does not contain a relevant part of the landfill, or it may contain another type of landfill given that most of the time different waste types are placed close to each other. If this is the case, the risk is that features of different landfills are brought close to each other in the embedding space, thus hurting the discriminative power of the downstream classifier which needs those representations to be far enough to differentiate among the two types of landfill. This can be accomplished by an

accurate choice of neighborhood size.

- **Distant tile selection**: the distant tile needs to be far enough from the anchor tile to be considered different. Thus, the distant tile center point is selected among the points for which the CAM is smaller than a given threshold. The lower the threshold, the more likely is to place the distant tile where there is no landfill. A higher threshold could allow considering another landfill as a distant tile. However, there is no guarantee that the two landfills would be of different types, thus hurting the learning of proper features. The distant tile can also be selected from an image that is not the one used for extracting the anchor and neighboring tiles.

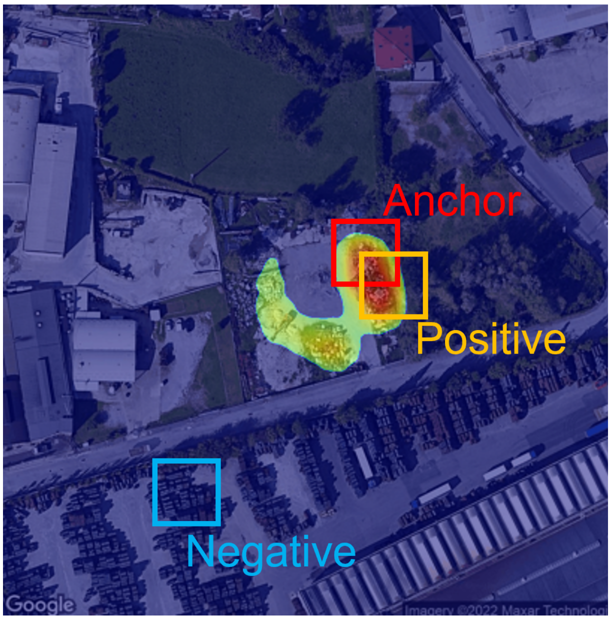


Figure 3.11: Example of how CAM-guided Tile2Vec should work. The anchor and neighboring (positive) tiles should be placed near each other where the CAM is high, whereas the distant (negative) tile should be placed where the CAM is lower.

The obtained results are very likely to be dependent on the goodness of the models that are used to produce CAMs. Following this approach, a data set of triplets can be created, and Tile2Vec can be used to learn a feature representation before performing TL.

Pseudo-label Generation

The CAMs generated by a classification model can be used not only to evaluate the localization capabilities of the network but also as a starting point for the design of WSL methods such as those explained in Chapter 2. However, while Chapter 2 describes a large number of sophisticated methods that can be exploited also in the RS domain, simpler approaches are considered given that the focus of this thesis.The basic idea is to generate segmentation masks as pseudo-labels that can then be used to train an instance segmentation network such as Mask R-CNN [52]. The simplest mechanism is based on a direct segmentation of the CAMs. In fact, the first step consists in generating CAMs for each image of the training set and for each class that is present in the multi-label GT. Then, the CAMs can be processed converting them into a segmentation mask by applying binarization based on a fixed threshold value. Finally, pseudo-labels in the form of polygons can be extracted from the contours of the binary CAMs. Alternatively, it is possible to exploit a more sophisticated approach, with the aim to refine CAMs and obtain pseudo-labels of higher quality. The considered approach is IRNet [228] which refines CAMs exploiting CRF [288] and taking into consideration the borders of the instances. In this case, the idea is to train IRNet using the CAMs generated by the classification model for the GT classes, and then use the trained model to generate the segmentation masks from which polygons can be extracted as in the previous case.

4 | Experiments and Evaluation

In this chapter, a description of the experiments and an evaluation of the obtained results are reported. Initially, an introduction to the evaluation procedure is provided. Then, the experiments are shown and evaluated according to the previously defined mechanisms. If needed, a description of the hyper-parameter tuning process for the different experiments is further provided. Once all the results have been highlighted, the best model is compared with the baseline and an analysis of the weaknesses and strengths of the model is carried out. To assess the possibility to proceed with a WSL task, segmentation masks are finally generated from the CAMs of the best-resulting model and analyzed to evaluate if they are good enough to use as pseudo-labels.

4.1. Evaluation Mechanism

To evaluate the results of the various experiments in the considered scenario, it is crucial to consider both a quantitative and a qualitative evaluation. A quantitative evaluation allows to obtain a quick idea of how well the classifier is performing from different perspectives. However, this information is not sufficient to understand why the model outputs certain predictions. For this reason, a qualitative evaluation is also carried out so that more understanding is gained of the actual capabilities of the classifier. All the following diagrams, plots, and analyses are generated using the evaluation tool named ODIN [19, 20], which was extended to match the analysis requirements of this thesis.

4.1.1. Quantitative Evaluation

To evaluate the results of the various experiments quantitatively, different metrics were considered. In general, *Accuracy* is one of the most intuitive metrics to evaluate a model. However, given that the data set is imbalanced, this metric is not suitable since high accuracy could be obtained without learning anything about the less-represented classes. For this reason, other metrics are employed. More specifically, each experiment is evaluated according to *Precision*, *Recall*, *F1-score*, and the *Precision-Recall curve*.

Given the presence of multiple classes, Precision, Recall, and F1-score are computed for each category in a one-vs-all fashion, meaning that a sample (or a prediction) is considered positive if the considered class is present (or predicted), whereas it is considered negative if the considered class is not present (thus, without taking into consideration the other classes). In this way, the metrics for each specific class are the same used in a binary scenario. Given this fact, it is possible to define the following terms:

- **True Positive** (TP): a sample is considered a TP if the considered class is present and predicted by the model.
- **False Positive** (FP): a sample is considered an FP if the considered class is not present but the model predicts it.
- **True Negative** (TN): a sample is considered a TN if the considered class is absent and the model does not predict it.
- **False Negative** (FN): a sample is considered an FN if the considered class is present but not predicted by the model.

Considering the above terms, the Precision, Recall, and F1-score for each class (c) can be computed by the following equations:

$$Precision_c = \frac{TP_c}{TP_c + FP_c}$$

$$Recall_c = \frac{TP_c}{TP_c + FN_c}$$

$$F1 - score_c = \frac{2 \times Precision_c \times Recall_c}{Precision_c + Recall_c}$$

The Precision metric indicates how many samples are correctly classified as containing a specific class (TP) among all the samples that are predicted as containing the class ($TP + FP$). Thus, high precision means that the network rarely predicts a class when it is not present. The Recall metric indicates how many samples are correctly classified as containing a specific class (TP) among all the samples that contain the considered class ($TP + FN$). Thus, a high recall means that the network is able most of the time to predict the presence of a class when it is present. The F1-score metric is a measure of the accuracy of the model that balances precision and recall.

Once the metrics for each class have been computed, it is possible to fuse them to obtain a high-level view of how the model is performing. In this study, the metrics are aggregated using a macro average which is simply the average of the considered metric for each class. Thus, *macro-Precision, macro-Recall, and macro-F1*, considering C different classes, can all be computed using the following formula:

$$macro - Metric = \frac{\sum_{c \in C} Metric_c}{|C|}$$

Besides these metrics, the Precision-Recall curve (PR curve) is taken into consideration. The PR curve shows the trade-off between Precision and Recall for different thresholds. The above metrics are computed considering a fixed threshold, meaning that all the samples whose classification score is above the threshold are considered positive, whereas the others are negative. In the PR curve, the threshold is not fixed instead. A high area under the curve represents both high recall and high precision. On the one hand, a model with high recall but low precision returns many results, but most of its predicted labels are incorrect compared to the GT labels. On the other hand, a model with high precision but low recall returns very few results, but most of its predicted labels are correct when compared to the GT labels.

4.1.2. Qualitative Evaluation

Even though the quantitative evaluation mechanisms described in the previous subsection allow to obtain an overall idea of how the model is performing, no indication is given on the aspects that are learned by the model. However, this is crucial, especially in the case in which the obtained results could be used for more complex tasks such as OD and IS. If more complex tasks need to be carried out on illegal landfills, it is not sufficient that the model can correctly predict the presence of, say, two classes in a sample. What is more important is that the model is actually able to distinguish among them and thus focuses on different relevant features. If this is not the case, detection tasks cannot be addressed successfully.

For this reason, all the experiments are analyzed qualitatively by looking at CAMs. More specifically, using the approach described in Section 3.2.3, a CAM is generated for each sample for each class. In this way, it is possible to observe where the model is focusing when predicting each specific class. In an ideal scenario, the model can focus on different aspects for every single class, meaning that it is able to successfully discriminate among the different categories.

4.2. Experiments

In this section, a description of the experiments that were performed in this study is reported. For each experiment, the obtained results are discussed and analyzed to check how to improve the results of the next experiments or to delineate possible future directions of improvement. When needed, the hyper-parameters tuning procedure is described. In the following, the experiments are reported in chronological order, from the first ones used to gain more insights on the data set, to those in which the techniques described in the previous sections are implemented and tested.

4.2.1. Training Environment and Configuration

All the experiments and models are implemented using the PyTorch framework[1]. The models are trained using 2 NVIDIA GeForce RTX 2080 Ti GPUs. If unless specified, the experiments were performed using almost the same parameters' values as Torres et al. [18]. More specifically, the batch size is set to 12 so that the full capacity of the GPUs is exploited if the non-cropped images are used. The learning rate is set to 0.005. The model is trained for 80 epochs using early stopping on the validation set to avoid overfitting. The metric for early stopping is the loss in case of binary classification or the macro-F1-score if multi-label classification is performed. This choice is dictated by the fact that F1-score allows giving importance also to less represented classes and can cope with data set imbalance. The early stopping patience is set to 8 with a min delta of 0.005. Finally, a threshold of 0.5 is used to distinguish between positive and negative classes.

4.2.2. Single-class Experiments

The initial experiments are performed considering only one class at a time solving the task of binary classification in which the aim is to predict the presence or absence of the considered class. The purpose of these experiments is to understand how much difficult it is for the network to recognize each class in the data set.

In this case, the data set is modified to make it compliant with the task that is being solved. More specifically, the multi-class multi-label annotations of the images are modified and transformed into binary labels. Considering a category c, if the original annotation contains class c (possibly together with other categories) the corresponding sample is labeled as positive; otherwise, it is labeled as negative. This operation is performed for each class, leading to the generation of a new binary data set for each class.

[1] https://pytorch.org/

Because of the way in which the binary data set is constructed, two types of negative samples are present:

- **Soft negative samples**: these are samples that also in the original data set were considered as negatives, meaning that they did not contain any type of landfill.
- **Hard negative samples**: these are samples that in the original data set contained one or more landfills' classes but not the classes considered in these experiments. In this study, these samples are called "hard" in the sense that, given that they contain other classes, they may be harder to classify.

While the number of soft negative samples is the same for all the classes, the number of hard ones depends on each specific class. However, the sum of the positive binary samples and the hard negative ones is always equal to the original number of positive samples.

Negative Sampling Strategy Selection

Considering the statistics of the AerialWaste data set reported in Section 3.1, it is rather intuitive that the obtained binary data sets are imbalanced since the number of negative samples is much higher than the positives. Thus, using these data sets can result in the classifier predicting the absence of the considered class. To avoid this problem, the binary data sets are modified by considering only a subset of the negative samples. The choice of negative samples can be a critical factor. For this reason, experiments considering the following negative sampling selection strategies are performed:

- **Random sampling**: this strategy selects negative samples randomly, without taking into consideration the type (soft or hard) of the sample.
- **Balanced strategy**: this strategy selects the same number of soft and hard negative samples. Moreover, the number of negatives is equal to twice the number of positives.
- **Imbalanced strategy**: this strategy selects the number of hard and soft negatives according to a given ratio with respect to the number of positives. The ratios that are taken into consideration are $[2:1, 2:0.5, 3:0.5]$. For instance, a ratio of $2:1$ means that the number of hard negatives is twice the number of positives and the number of soft negatives is equal to the number of positives.

To verify which of the previously defined strategies allows to obtain a more discriminative classifier, a binary experiment is conducted for each class and for each of the considered strategies (if enough samples are available). Concerning the data set, the original training set is split into training and validation sets, keeping 80% of the samples for training and the remaining ones for validation. Then, once the split is performed, the original data set is transformed into a binary data set for each class following the described procedure. The number of negative samples in each binary data set is reduced to match the chosen strategy. This is accomplished by taking a random subset of the needed soft and hard negative samples.

At this point, all the experiments are run employing the training configuration described in the previous subsection, and, for each class, the models are compared quantitatively and qualitatively, thanks to the use of CAMs, on the validation set. Each network is pre-trained using ImageNet pre-training. From a quantitative point of view, the balanced strategy allowed to obtain better results than using a random strategy. This can be due to the fact that random sampling produced data sets in which the number of hard negatives is much more reduced than the number of soft negatives. It is rather intuitive that hard negatives are more relevant than soft ones since they allow to understand whether the network is able to comprehend if the class is present even when other potentially similar instances are given. Considering imbalanced strategies instead, no major improvement is obtained. This can also be due to the limited number of samples available and the complexity of the task. Moreover, for many of the classes (e.g., *Glass*, *Asphalt milling*), the network tends to always predict the absence of the class. This can be led back to the very limited number of samples of the considered classes.

From a qualitative point of view, it is possible to verify that, independently of the class, the network does not focus only on illegal landfills. Once again, this can be brought back to the limited number of samples, at most around 400 (*Heaps not delimited*).

Transfer Learning Strategy Selection

Given that the previous models are not able to focus on illegal landfills and that this is a core point for the success of the task, the possibility of performing TL using the weights of the model provided by Torres et al. [18] was tested. From now on, this is referred to as Torres-AerialWaste pre-training.

For each class, an experiment was conducted using Torres-AerialWaste pre-training and fixing the negatives samples selection to the balanced strategy. This time, from a qualitative point of view a major improvement was obtained since the models are able to focus on illegal landfills. In this case, the source and target tasks are much more similar. However, every model focuses only on generic illegal landfills, without being able to distinguish which portion of the landfill contains the considered class. Moreover, from a quantitative point of view, it is possible to assess that either the model always predicts the class as absent (if limited samples are available) or the number of FPs is very high. This can be brought back to the fact that the models are only able to find the presence of a landfill but cannot understand its type. The same experiments were performed using the random strategy for the selection of negative samples, once again testifying the superiority of the balanced strategy from a quantitative point of view.

Data Set Augmentation

Given that the above results can also be related to the limited number of available samples, the possibility of increasing the size of the data set is considered. In particular, to generate more samples, offline data augmentation as described in the previous chapter is performed. More specifically, given a crop size, the original images are cropped into five partially overlapping patches, each labeled in the same way as the original image. In this way, the training set of each binary experiment is five times larger.

Fixing the negative sampling strategy to the balanced strategy and performing the Torres-AerialWaste pre-training, different experiments were conducted for each class using various crop sizes. More specifically, the analyzed crop sizes are $[500, 600, 700, 800]$. This choice is dictated by the fact that most original images are 1000×1000 pixels. Thus, the crops obtained using crop size 500 do not overlap (except for the center patch), while in the other cases, the overlap is partial. Considering smaller sizes would not be convenient since the risk of not including a landfill in the patch is higher. Moreover, if the image size is smaller than 800×800 pixels, only the center crop is performed to avoid using samples that are too similar to each other.

The results show that, especially from a qualitative point of view, offline data augmentation leads to an improvement. This proves the fact that increasing the number of samples, improves the capabilities of the classifiers. Concerning the crop size, for most of the classes, the best results were obtained using a crop size of 600 pixels. A crop size of 700 pixels often provides similar results and improves especially in the case of *Rubble-excavated earth and rocks*. Instead, considering smaller (less than 600) or bigger (more than 700) crop sizes results in a degradation of the performance. The reason can be due to the fact that, on one side, a small crop size increases the risk of introducing too many mislabeled samples which hurts the discriminative power of the classifier. On the other side, a big crop size introduces much more context that may cause more confusion for the classifier.

Final Binary Assessment

Given the above results, a final set of binary experiments is performed:

- The crop size is fixed to 650 pixels since the results provided using a crop size of 600 and 700 were similar.
- The negative sampling strategy is fixed to balanced.
- Both ImageNet pre-training and Torres-AerialWaste pre-training are tested.

The results confirmed the previous considerations, meaning that ImageNet pre-training tends to perform worse than Torres-AerialWaste pre-training. Moreover, most of the time the focus of the models is only on illegal landfills, without any differentiation on the actual type. Classes poorly represented, such as *Glass, Asphalt milling, Corrugated sheets* are never detected given that the number of samples, even after augmentation, is still too limited.

In the case of more represented classes such as *Rubble, Bulky items* and *Heaps not delimited* it is possible to observe the CAMs and notice that when two or more types of landfills are present, the network focuses on different aspects to predict the presence of each class (in the corresponding binary experiment). Figure 4.1 shows an example of this behavior.

4.2.3. Multi-label Experiments

Thanks to the results of the single-class binary experiments, it is possible to set up a framework for the multi-label experiments. The experiments follow a pre-defined procedure: 1) a baseline experiment is chosen, 2) a set of experiments is defined to improve the results with respect to the baseline using the methods described in Section 3.3, 3) all the results are evaluated quantitatively and qualitatively.

Multi-label Data Set

Starting from the AerialWaste data set and taking into consideration the results of the single-class experiments, it is possible to select a subset of classes following the principle that if it is not possible to find and recognize a class in a single-class experiment, it is more difficult to do it in a multi-label experiment. Moreover, given that for some classes the number of available images is much reduced, it is very unlikely that a good model can be derived using such a limited amount of data. Thus, some classes are discarded.

Since a potential source of confusion is that *Waste types* and *Storage modes* are not

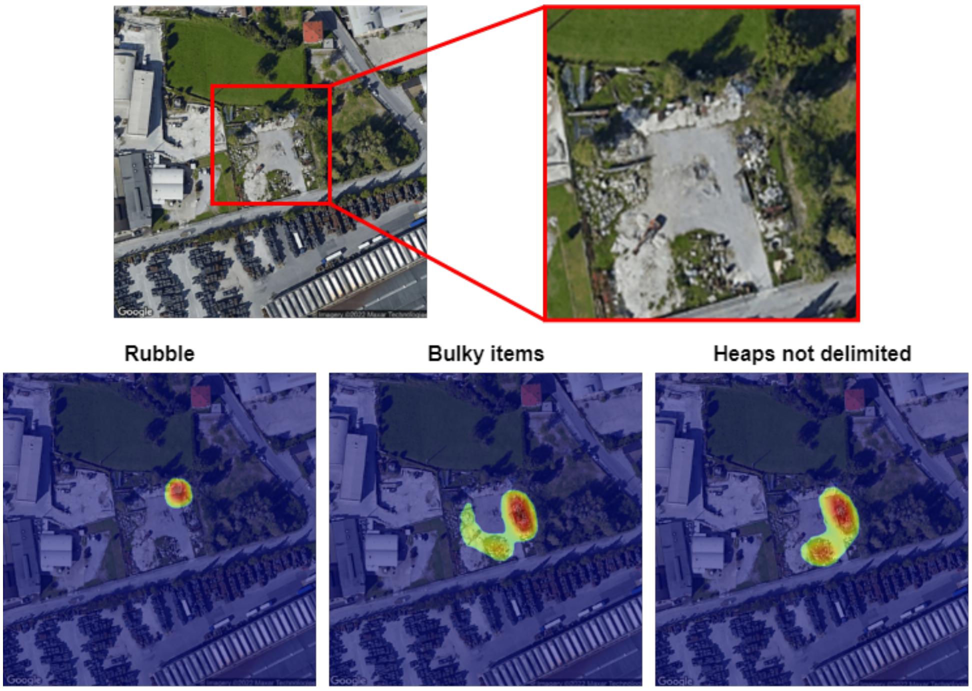


Figure 4.1: Example of an image from the AerialWaste data set [18] classified as TP by three different binary classifiers. As it is possible to see the CAMs focus on different aspects in the 3 cases, which are correct.

mutually exclusive, given that the storage mode indicates a container of waste, whereas the waste type is related to the content, most of the time a waste instance can be classified according to two different views: container and content. For this reason, only the classes indicating *Waste types* are kept, leading to a final set of 5 categories: *Rubble, Bulky items, Fire Wood, Scrap, Vehicles*. Given the selected classes, it is possible to construct the data set used for all the subsequent experiments: each sample is considered positive if it contains at least one of the five selected classes and negative otherwise.

The previously introduced concept of hard and soft negatives can be generalized to the case of multi-label classification. Given that only *Waste types* are taken into consideration, hard negatives are selected among negative samples that originally contained at least one waste type. This results in having a very limited number of hard negatives with respect to the number of soft negatives. However, in the following experiments, all these samples are kept because of their importance as already highlighted during the binary experiments. Concerning the soft negatives, preliminary experiments revealed that using a number of negative samples equal to 25% of the positive samples seems to be the most promising direction. The resulting data set is defined as follows:

- A trainval set derived from the AerialWaste training set, containing 418 positive samples, 36 hard negatives, and 104 soft negatives, for a total of 558 samples.

- A test set derived from the AerialWaste test set, with 135 positive samples, 33 hard negatives, and 33 soft negatives, for a total of 201 samples.

Since this data set is derived from the original AerialWaste data set, pixel-level annotations are available only for the test set. The test set is used only for the evaluation of the final selected model.

To perform the various experiments, the trainval set is further split into training and validation sets, keeping 80% of the samples for training and 20% for the validation set that is used for early stopping and to compare the different models. The splitting operation is uniform, meaning that the samples are not randomly selected, but instead, the selection is performed in such a way that the distribution of the classes is left unmodified. Thus, the training set contains 80% of the soft negatives, 80% of the hard negatives, and 80% of the samples of each considered class.

After the split, the resulting training and validation set are as follows:

- The training set contains 334 positive samples, 30 hard negatives, and 83 soft negatives, for a total of 447 samples.
- The validation set contains 84 positive samples, 6 hard negatives, and 21 soft negatives, for a total of 111 samples.

Because of the limited dimension of the training set and the considerations derived from the single-class experiments, the training set is enlarged using offline data augmentation as described in Section 3.3: the crop size is set to 650 as for the final single-class experiments. Following the same strategy adopted during the single-class experiments, if the image size is smaller than 800 × 800 pixels, only the center crop is performed.

This results in the final adopted multi-label data set composed by:

- A training set with 1,482 positive samples, 146 hard negatives, and 391 soft negatives, for a total of 2,019 samples.
- A validation set with 84 positive samples, 6 hard negatives, and 21 soft negatives, for a total of 111 samples.
- A test set with 135 positive samples, 33 hard negatives, and 33 soft negatives, for a total of 201 samples.

After these operations are performed on the data set, it is crucial to check if the characteristics of the data set described in Section 3.1 are left unmodified. By looking at

Table 4.1: Number of images for each selected class in the final data set. Both the number of samples into training, validation and test set, and the total are reported.

Label	#Training samples	#Validation samples	#Test samples	#Samples
Rubble/excavated earth and rocks	786	46	67	899
Bulky items	846	48	44	938
Fire Wood	480	27	38	545
Scrap	468	28	27	523
Vehicles	106	5	27	138

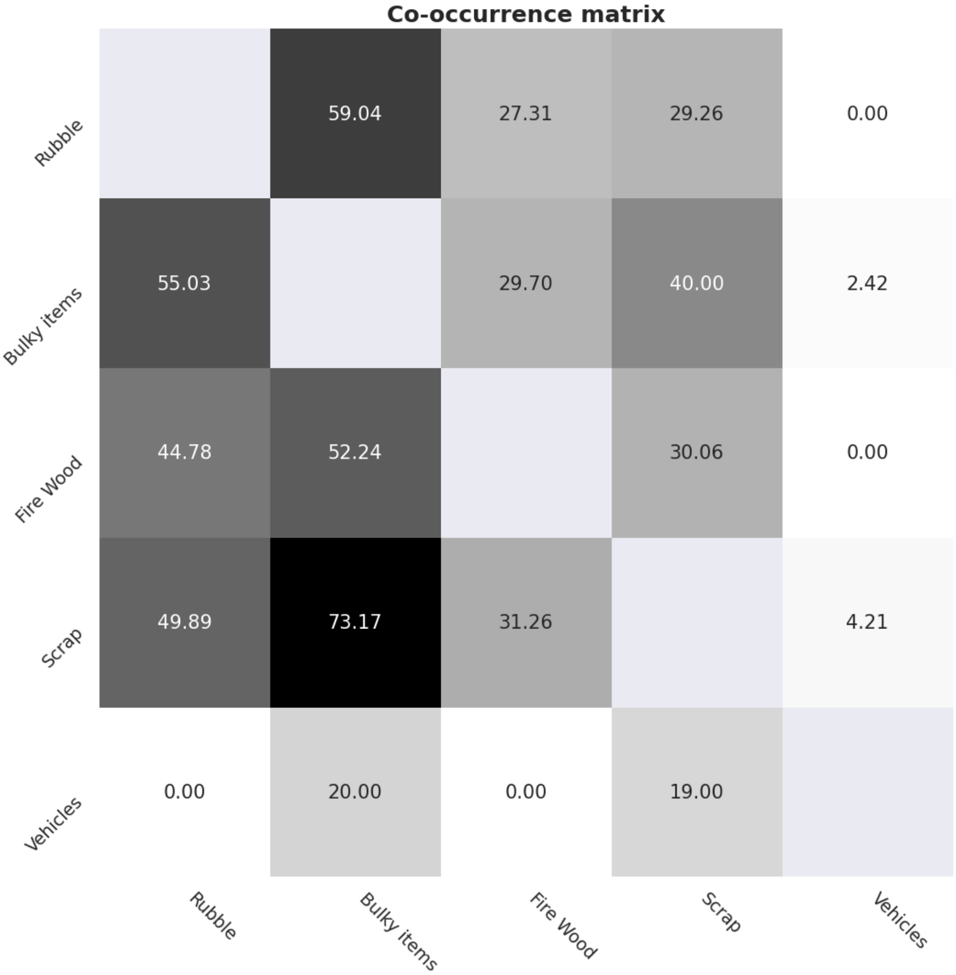
the second column of Table 4.1, it is evident that the data set is still imbalanced. The most represented class (*Bulky items*) is present in 8 times more samples than the least represented one (*Vehicles*).

Moreover, as shown in Figure 4.2, the high class co-occurrence in the training set is still an issue that needs to be tackled in the various experiments, using approaches similar to those described in Section 3.3.

Furthermore, analyzing the distribution of the positive samples in the training set, based on the number of classes that are present in each sample, it is possible to attest that the average number of samples gradually decreases as the number of classes present in the samples increases (top-left image of Figure 4.3), e.g. 33.3% of the samples contain 1 class while only 4.5% of the samples contain 4 classes. Unfortunately, from Figure 4.3) we can also observe that, for almost all classes, the number of samples with only a specific class is very low while two or more classes are present most of the time. In this case, it can become harder for the classifier to learn to discriminate classes due to the high confusion that can be introduced. The most critical class is *Scrap*: the number of times it appears alone is lower than the number of times it is present with other 1, 2, and 3 classes.

It is important to clarify that, even if a sample is labeled as containing only one class, it is still possible that other classes aside from those selected for classification, e.g., *Tires, Asphalt milling, Plastic* are present. This leads to a sort of *hidden co-occurrence* that is hard to detect but that can potentially further complicate the classification task. Hard negatives should help in solving this issue.

In addition to the three sets (training, validation, and test) defined for the supervised experiments, a large number of data with coarse-grained annotations is available. In particular, the original AerialWaste data set contains 2,065 binary annotated images in which at least an illegal landfill is present but lacks fine-grained annotations indicating the specific type of landfill. This set of images can be split into a training set and a validation set , with respectively 1,652 and 413 samples, to be used in a self-supervised



	Rubble	Bulky items	Fire Wood	Scrap	Vehicles
Rubble		59.04	27.31	29.26	0.00
Bulky items	55.03		29.70	40.00	2.42
Fire Wood	44.78	52.24		30.06	0.00
Scrap	49.89	73.17	31.26		4.21
Vehicles	0.00	20.00	0.00	19.00	

Figure 4.2: Relative co-occurrence matrix of the waste types classes of the training data set used in multi-label experiments.

scenario.

Baseline

During the analysis of single-class experiments, a set of considerations concerning the backbone pre-training and the characteristics of the training set were analyzed. Based on what was observed during these experiments, the Torres-AerialWaste pre-training was considered as the way to initialize the weights of the ResNet50 backbone given that it provided better results with respect to ImageNet pre-training. Moreover, the training data set is cropped to a square of size 650 pixels.

This configuration is used to execute a first multi-label classification experiment on the multi-label data set. Analyzing the obtained results, it is possible to verify that considerations similar to the single-class experiments can be drawn. In fact, from a

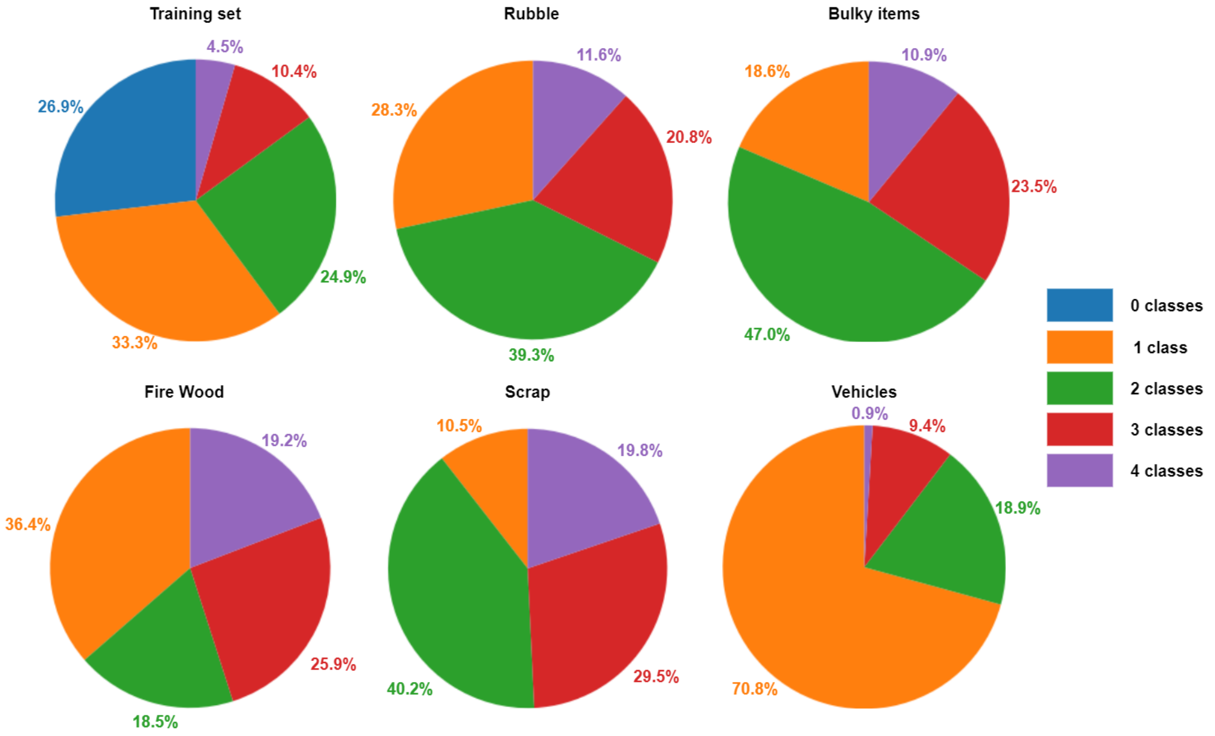


Figure 4.3: Distribution of the number of samples according to the number of classes contained in each sample.

quantitative point of view, this experiment allows reaching around 57.15% of F1-score (4.2). However, if the results are analyzed more in-depth, it is possible to see that *Rubble* and *Bulky items* reach 57.97% and 76.52% of F1-score respectively, while *Fire Wood* and *Vehicles* are below 50%.

However, from a qualitative point of view, it is possible to realize that the model often confuses one class for another without discriminating enough between them. This results in predicting the presence of multiple classes even when only a few are present given that there is no real understanding of the differences between the classes. This problem is likely to be due to the characteristics of the data set and specifically to the high co-occurrence of the classes that makes it difficult for the model to learn what a certain class looks like given that it appears too many times with other classes. In addition, inter-class similarity and intra-class diversity of the samples can contribute to the further increase of confusion.

Oversampling

Given the issues present in the baseline experiment, the idea is to apply oversampling as described in Section 3.3 to reduce the problems of high class co-occurrence and data set imbalance. Given that oversampling is applied by creating copies of the images already

Table 4.2: Quantitative evaluation of the models obtained in the various experiments on the validation set.

Experiment	Metric	Rubble	Bulky items	Fire Wood	Scrap	Vehicles	Macro avg.
Baseline	*Precision*	86.96%	65.67%	88.89%	41.67%	66.67%	69.97%
	Recall	43.48%	91.67%	29.63%	89.29%	40.00%	58.81%
	F1-score	57.97%	**76.52%**	44.44%	56.82%	50.00%	57.15%
Oversampling (Simplex-guided)	*Precision*	65.12%	67.92%	52.17%	50.00%	100.00%	67.04%
	Recall	60.87%	75.00%	44.44%	32.14%	40.00%	50.49%
	F1-score	62.92%	71.29%	48.00%	39.13%	57.14%	55.70%
SDA (Uniform strategy)	*Precision*	71.05%	62.32%	52.38%	40.00%	100.00%	65.15%
	Recall	58.70%	89.58%	40.74%	35.71%	40.00%	52.95%
	F1-score	64.29%	73.50%	45.83%	37.74%	**57.14%**	55.70%
SDA (Uniform strategy + Blur)	*Precision*	60.00%	68.33%	50.00%	48.57%	100.00%	65.38%
	Recall	58.70%	85.42%	29.63%	60.71%	40.00%	54.89%
	F1-score	59.34%	75.93%	37.21%	53.97%	**57.14%**	56.72%
SDA (Simplex-guided + Blur)	*Precision*	56.92%	60.00%	53.13%	41.27%	100.00%	62.26%
	Recall	80.43%	87.50%	62.96%	92.86%	40.00%	72.75%
	F1-score	**66.67%**	71.19%	**57.63%**	**57.14%**	**57.14%**	**61.95%**

part of the data set, the Simplex-guided strategy is employed. More specifically, the parameters of the strategy are set in such a way that the relative co-occurrence between every two classes must be smaller than 30% ($k_{i,j} = 0.3 \quad \forall i \neq j$) and that the maximum imbalance between the number of samples of two classes is 20% ($b_{i,j} = 0.2 \quad \forall i \neq j$). Finally, all the samples from the original data set are also kept ($u_i = 0 \quad \forall i$). In this case, the distribution of the samples based on the number of classes, present in an image, is not forced with any specific constraint. After executing the Simplex Algorithm, 2,771 new samples are added to the original training set. More specifically, Simplex outputs the number of samples that should be added to the data set for each possible subset of classes. Thus, images are randomly sampled to meet the required Simplex output.

Observing the results reported in Table 4.2, it is possible to verify that *Bulky items* is still the class with the highest value for the F1-score. Concerning the other classes, the F1-score is more balanced between classes than in the baseline experiment. This can be caused by the added samples. This results in a model that is less prone to predict the presence of many classes. Overall, from a quantitative point of view, the average F1-score is 55.70%. From a qualitative point of view, the results are similar to the baseline, but in some cases, it is more evident that the model is trying to differentiate among the different classes. Most of the time, the number of predicted classes is reduced, whereas, in the case of the baseline, many classes were most of the time predicted together incorrectly. This shows the importance of acting on the class co-occurrence and imbalance to increase the discriminative power of the classifier. However, using only oversampling of the available samples does not bring a major improvement.

Uniform Augmentation with Synthetic Data

The oversampling experiment confirms the importance of enlarging the size of the training set while tackling the problem of co-occurrence and imbalance. The Simplex guided strategy is quite trivial for oversampling since the selected samples for the augmentations are taken from the original ones, and there is no risk of including unrealistic samples. The scenario is different when synthetic data augmentation (SDA) is used. Generating synthetic data, as described in 3.3, with the possibility of adding multiple classes to the same sample, could lead to very unrealistic samples. For this reason, in the experiments, only samples containing instances of a single class are generated. The problem with this strategy is the need for a larger amount of samples to reduce the co-occurrence.

Several experiments were conducted considering the addition of a different number of synthetic samples for each class and a different number of instances for each generated sample. More specifically, the number of newly created samples per class was chosen between $[187, 375, 750]$ while the number of instances per sample was randomly picked from one of the ranges $[1, 2], [2, 4]$. Most of the time, there is no big difference on average from a quantitative point of view, but using a smaller number of instances (in the range $[1, 2]$) produced models that are less able to differentiate among the classes. Reducing also the number of generated samples per class resulted in a reduction of the qualitative outcome of the model as shown in Figure 4.4. More specifically, using 187 samples per class resulted in obtaining over 60% F1-score but the network did not learn anything about the different classes and is thus unable to differentiate among them. This can be due to the fact that introducing 187 samples per class does not reduce enough the co-occurrence and imbalance of the classes. Overall, the best experiment is the one that adds 750 synthetic samples for each class, each with a number of instances in the range $[2, 4]$, for a total of $3,750$ samples added to the original training set of $2,019$ samples. This results in effectively reducing the relative co-occurrence given that a lot of samples with only a single class are present in the final data set. Moreover, the relative imbalance is also reduced given that $\frac{x+750}{y+750} \ll \frac{x}{y}$ where x and y are the numbers of original samples of two different classes.

One of the strengths of synthetic samples is that they cannot be mislabeled. However, synthetic samples are often unrealistic and, because of the limited amount of patches used to generate them, the network could overfit. This can be partially mitigated using flip and rotation augmentations. The best generation strategy was tested with the addition of techniques to improve the realism of the generated images. More specifically, experiments blurring the contours and using a CAM-guided approach, are carried out. Despite the CAM guidance allowing the creation of impressively realistic images, the best overall

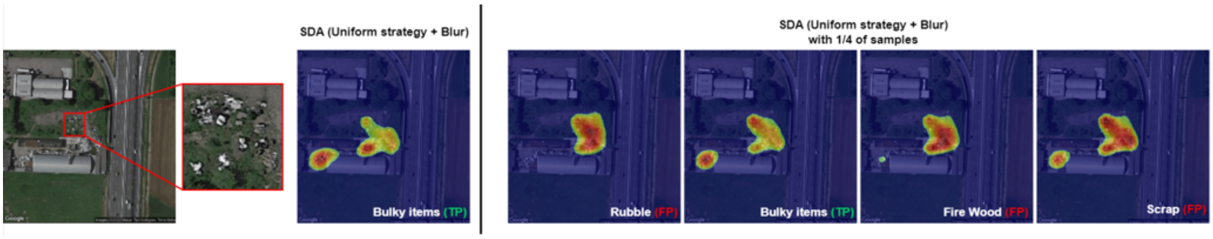

SDA (Uniform strategy + Blur)
Bulky items (TP)
SDA (Uniform strategy + Blur) with 1/4 of samples
Rubble (FP)
Bulky items (TP)
Fire Wood (FP)
Scrap (FP)

Figure 4.4: Qualitative comparison of SDA with 750 samples per class and SDA with 187 samples per class on an image from the validation set. As can be seen on the right, using fewer samples per class causes the network to make more confusion between classes.

results are obtained using random positioning of the patches. From a quantitative point of view, using SDA allows to reach an average F1-score of 55.70% without blur and 56.70% using blur, showing comparable results with the oversampling and slightly worse than the baseline. An in-depth analysis of the experiments with and without blur reveals that the behavior is very similar from a qualitative point of view. However, blurring the contours allows obtaining better results in the case of *Scrap*, whereas it worsens the performance for *Fire Wood*. This can be due to the fact that this class is often already similar to the background. Thus, blurring the contours may make it even more difficult to detect for the network, explaining why the network learned more about it without blur. This does not hold for *Scrap* whose added patches are already much more distinguishable from the background. More details about the quantitative results are reported in Table 4.2 showing performance over the classes. The same considerations made for oversampling hold. However, from a qualitative point of view, even though the baseline obtains an F1-score greater than the uniform SDA, a qualitative analysis shows that the model generates more confusion and is less able to differentiate between classes, as reported in Figure 4.5.

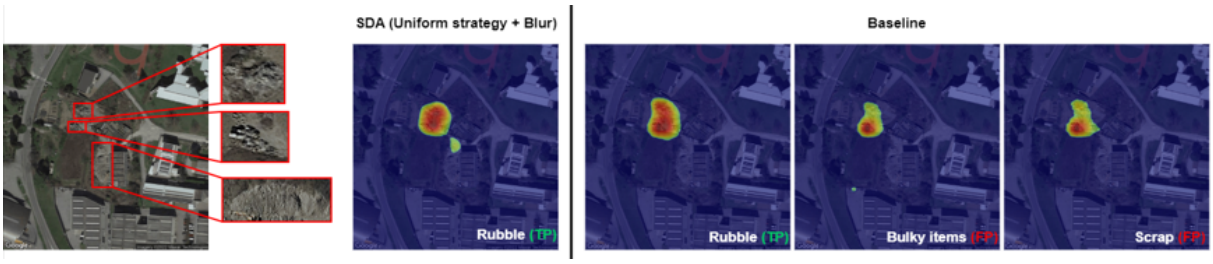


Figure 4.5: Qualitative comparison of SDA with uniform strategy and the baseline on an image from the validation set. The baseline is less able to differentiate among the different classes, resulting in more confusion.

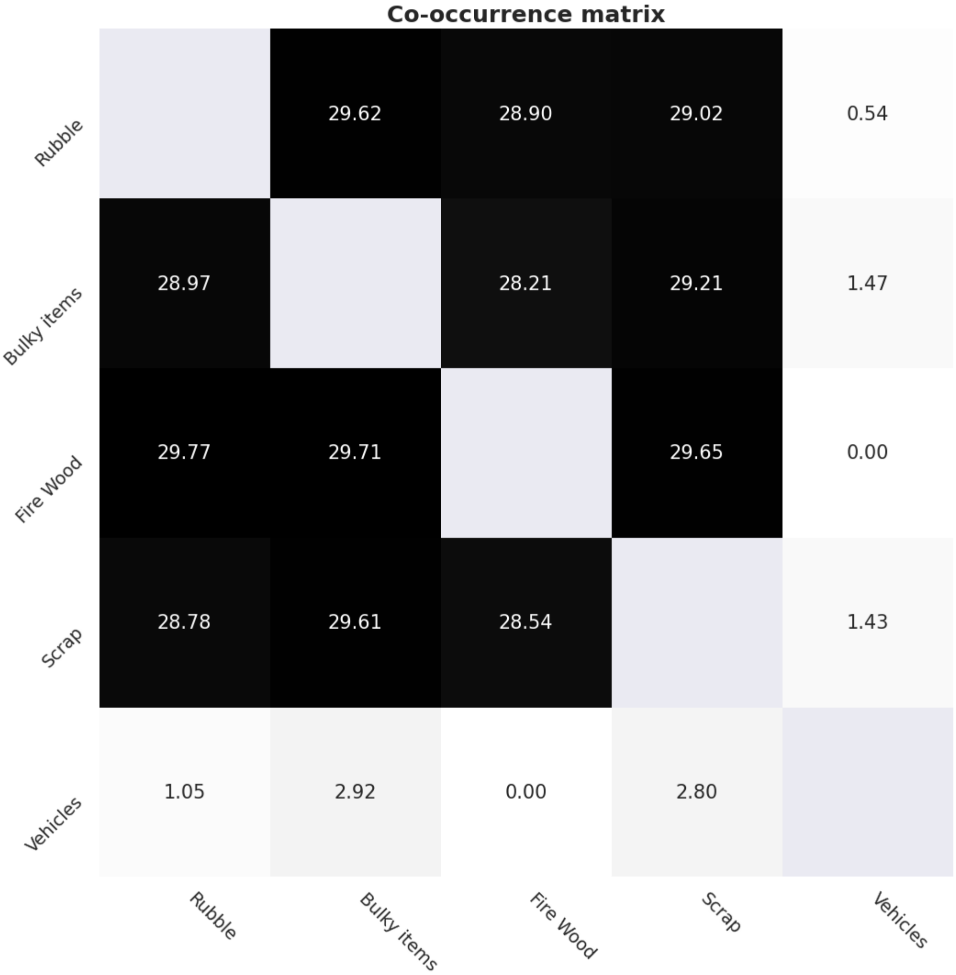
Table 4.3: Number of images for each selected class in the training data set and in the training set augmented with the Simplex-guided augmentation strategy. In the original data set the relative difference between the most represented class (*Bulky items*) and the less represented one (*Vehicles*) is around 80%, while in the augmented data set, the relative difference between the same classes is around 50%.

Label	#Original training samples	#Augmented training samples
Rubble/excavated earth and rocks	786	1685
Bulky items	848	1726
Fire Wood	480	1630
Scrap	468	1699
Vehicles	106	863

Simplex-guided Augmentation with Synthetic Data

Given the improvement obtained using SDA, a more sophisticated strategy for generating samples is experimented. In particular, a Simplex-guided strategy is taken into consideration forcing the relative co-occurrence of the augmented data set to be smaller than 30% for each pair of classes ($k_{i,j} \leq 0.3 \quad \forall i \neq j$). Furthermore, the maximum relative imbalance between every two classes is at most 50% ($b_{i,j} = 0.5 \quad \forall i \neq j$). Finally, another constraint is set considering the number of classes present in each sample: the maximum percentage difference between the number of samples with k classes and the number of samples with $k + 1$ classes is set to $+5\%$ for each class ($d = 0.05$). The parameters have been chosen to introduce around the same number of samples but allowing the generation of synthetic data with multiple classes being sure that the co-occurrence and imbalance are still constraints to be reduced. Using this strategy, 3,768 samples are generated using random patch positioning and a number of instances per sample in the range $[2, 4]$ (in case only one class is added) or in the range $[1, 2]$ (if two or more classes are added). This is done to reduce the number of instances inserted in each sample, which could cause much more confusion given the addition of multiple classes.

Table 4.3 shows the effect of the augmentations on the number of samples per class and on their imbalance. Figure 4.6 shows the effect on the relative co-occurrence matrix, and Figure 4.7 shows the effect on the distribution of the number of samples according to the number of classes.



	Rubble	Bulky items	Fire Wood	Scrap	Vehicles
Rubble		29.62	28.90	29.02	0.54
Bulky items	28.97		28.21	29.21	1.47
Fire Wood	29.77	29.71		29.65	0.00
Scrap	28.78	29.61	28.54		1.43
Vehicles	1.05	2.92	0.00	2.80	

Figure 4.6: Relative co-occurrence matrix of the augmented training set obtained using the Simplex-guided augmentation strategy with a maximum allowed co-occurrence of = 0.3.

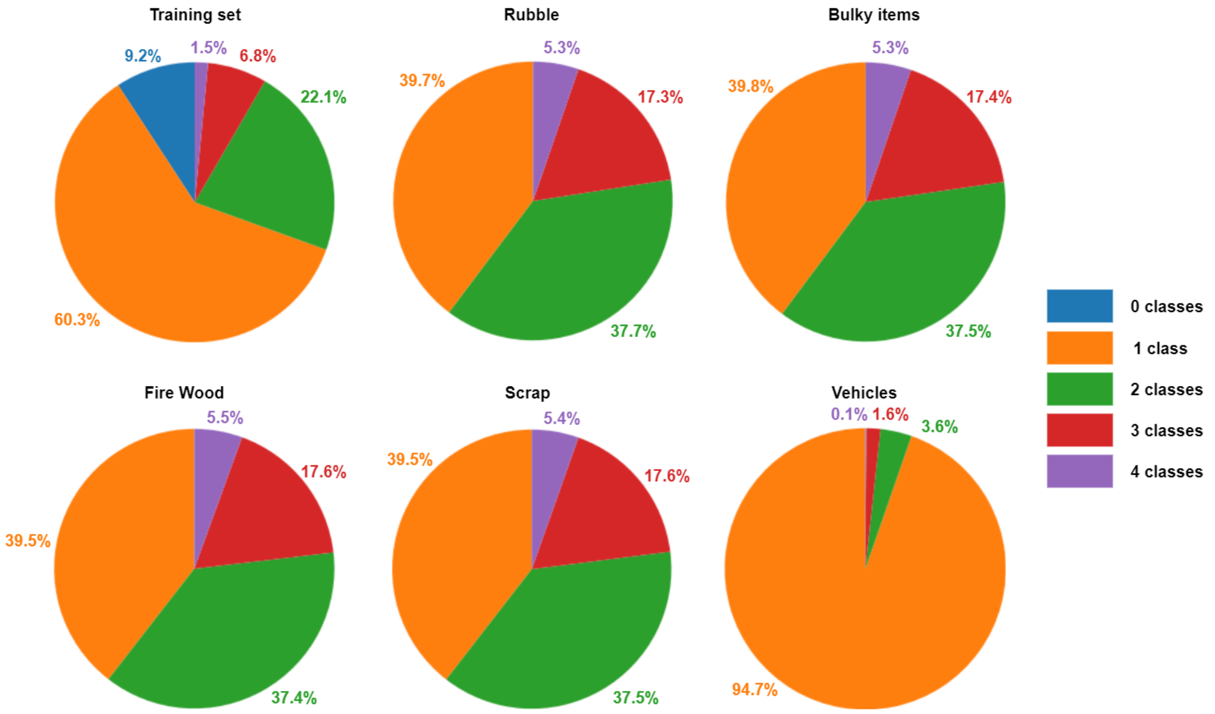


Figure 4.7: Distribution of the number of samples according to the number of classes contained in each sample for the augmented training set obtained using the Simplex-guided augmentation strategy.

The obtained quantitative results are better than the previously described experiments and are reported in Table 4.2. Contours blurring is used. Even though from a quantitative perspective the Simplex-guided model is the best, from a qualitative perspective the model is less able to differentiate between classes, similar to the baseline, as shown in Figure 4.8. From a qualitative point of view, the uniform augmentation-based model is better. It is important to notice that, in this case, a quantitative improvement of the performance does not imply that the network is able to discriminate better between the five selected classes. In fact, from the previous experiments, if a model almost always predicts that three or more classes are present in an image, it can obtain a better F1-score with respect to a model that is less confident but has learned more accurate features. Once again, this issue is related to the high class co-occurrence. Thus, given the importance of differentiating between the different categories, from a localization point of view, uniform SDA is chosen as the model for the final evaluation of the test set. Moreover, given that blurring the contours provides a slight improvement to the F1-score, this model is preferred.

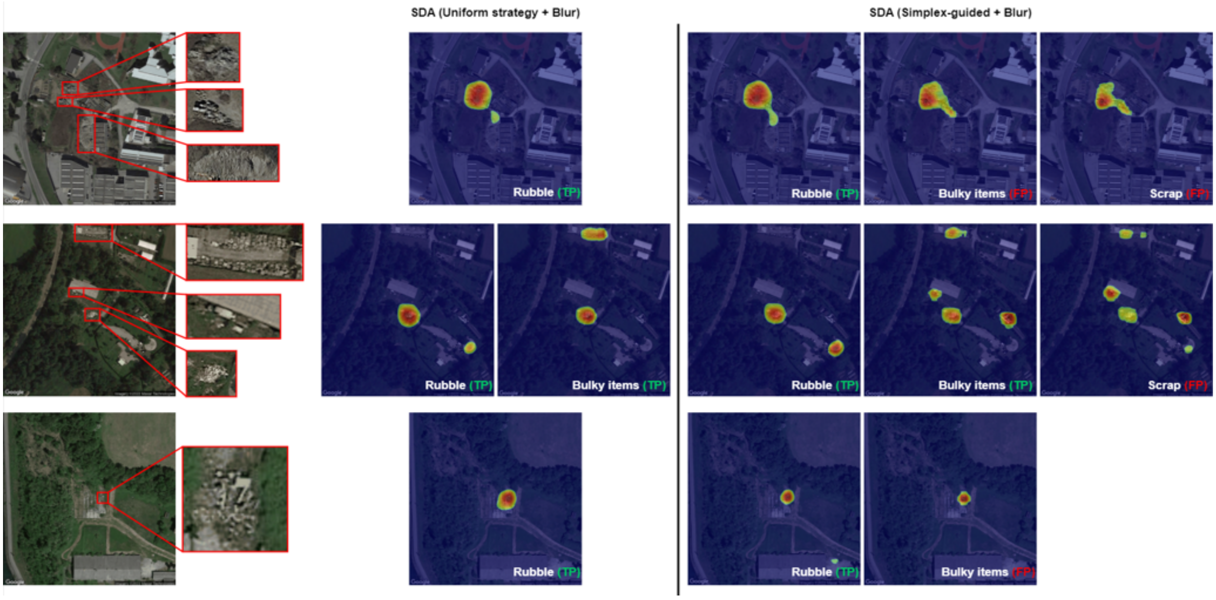


Figure 4.8: Qualitative evaluation of the Simplex-guided SDA and comparison with SDA on a few images from the validation set.

Self-Supervised Learning

The large number of samples not annotated with multi-label annotations (2,065) cannot be exploited to train the network in an FS fashion. However, the unlabeled training and validation sets can be used to perform SSL, as described in 3.3.

The potential advantage of using SSL is related to the possibility of enhancing the feature representations in such a way that they can carry a greater discriminative power. In this scenario, the large amount of images with only binary labels is helpful to keep only the samples with an annotated landfill and discard the negatives that do not contain any useful feature to learn. It is important to notice that SSL does not require binary labels for the training of the pretext task. As already highlighted in Section 2.3, in absence of an adequate pretext task, the learned features are likely to be useless and may result in performance deterioration.

Considering the best model obtained so far (SDA with uniform strategy, random positioning of the patches, and contours blurring), different weights initialization are taken into consideration for the backbone, using the weights obtained by training a pretext task in a self-supervised manner. More specifically, the considered self-supervised approaches are those highlighted in Section 3.3, i.e., predicting image rotations, solving jigsaw puzzles, and Tile2Vec. The results of the different experiments are reported in Table 4.4. Since the first two tasks are predictive pretext tasks, it is possible to exploit regularization strategies such as early stopping to avoid overfitting. Thus, these pretext tasks are trained using only the training set, while the validation set is used for validation. Regarding Tile2Vec, instead, given that a contrastive learning framework is used, no validation is considered and thus, the full set of trainval data is used for training.

To obtain the final weights, the first two pretext tasks (proposed in [76] and [99] are trained for 80 epochs using early stopping to prevent overfitting based on the loss value, with 8 epochs of patience and a min delta of 0.005. The learning rate is set to 0.005. Moreover, the images are cropped in the center randomly using a crop size of 600, 800, or 1000 pixels and then rescaled to 800 × 800 pixels to ensure that the size is the same for all the images in a batch. This is used to include different amounts of context during training. Moreover, concerning the pretext task of solving jigsaw puzzles, a patch of size 255 × 255 is extracted from the rescaled image and divided into 4 tiles of size 64 × 64 pixels. After training the pretext tasks, the downstream task training is performed. Different experiments are performed with and without freezing the first two stages of the backbone during the pretext task training. For both the pre-training techniques, not freezing the first two stages provided better results. However, the overall performance is comparable to the ImageNet pre-training and worse, especially from a quantitative point of view, than the case in which the weights are initialized using the Torres-AerialWaste pre-training. This can be due to the fact that as reported in Chapter 2, the pretext task based on image rotations may not be suitable in the case of RSIs where it is difficult to define an orientation, whereas the one based on solving jigsaw puzzles may fail due to the fact that in RSIs, the spatial correlation is not so strong as in natural images.

Concerning Tile2Vec, the CAM-guided approach defined in the previous chapter is employed given that randomly selecting tiles does not guarantee to select portions of the images containing landfills, leading to no relevant knowledge gain for the downstream task. For all of the experiments, the CAM-guidance was conducted using the binary model published by Torres et al. [18]. While the training configuration is the same as in the previous cases, a set of parameters specifically related to Tile2Vec needs to be set:

- **Classification threshold**: this parameter indicates the minimum confidence for the selection of the images from which tiles are extracted. If the binary model classifies the image as positive with a value higher than this threshold, the image is considered for patch extraction otherwise it is discarded.
- **CAM positive threshold**: this parameter indicates the minimum value that a pixel in the CAM should possess to be selected as the center point for the extraction of the anchor tile.

- **CAM negative threshold**: this parameter indicates the maximum value that a point in the CAM should possess to be selected as the center point for the extraction of the distant tile.

In particular, different combinations of these parameters are tested and the quality of the produced patches is analyzed before training. The best tiles are obtained using a classification threshold of 0.50, a CAM positive threshold of 0.8, and a CAM negative threshold of 0.40.

Training Tile2Vec and then transferring the weights for the downstream task resulted in poor performance. The average F1-score is, in fact, around 7% smaller than that obtained using ImageNet pre-training and around 10% smaller than that obtained using Torres-AerialWaste pre-training. This can be due to the fact that selecting a neighbor so close to the anchor does not allow to account for intra-class diversity. At the same time, selecting the distant tile where the CAM value is low does not allow to account for inter-class similarity. However, these aspects are fundamental to learn discriminative features. Starting from this consideration, an interesting future direction could be to sample hard neighboring and hard distant tiles. For instance, this can be done if there already exist models that are able to recognize a specific class. In this way, it is possible to improve the learned features by selecting the neighboring tile also from an image that is different from that used for the anchor. If only an image is used, the neighboring and anchor tiles look quite similar. Instead, if two images are used, it is possible that the tiles are more different (e.g., due to high intra-class diversity) but the learned features may be more relevant thanks to the diversity of the samples. Furthermore, if models for different classes are available, it is possible to select distant tiles choosing also among the anchors of other classes. This way, the anchor and neighboring tiles would contain a specific class, while the distant one would contain a different one. In this way, the training of the pretext task can potentially allow to:

- Pull instances of the same class close to each other in the feature space.
- Push instances of different classes far from each other in the feature space.
- Push instances of landfills far from tiles representing the background in the feature space.

This approach has not been tested because the discriminative performances reached by the described binary classifiers are lower than those required to proceed with this CAM-guided Tile2Vec variant. However, if more focus is given to developing single-class classifiers, then using the described approach may improve the results for multi-label classification thanks to the exploitation of unlabeled data which often provides better feature representations.

Table 4.4: Quantitative evaluation of the models obtained in the various experiments on the validation set. All these models are trained using the best SDA strategy from the previous experiments. The difference between them is in the backbone's weights initialization, that follows different strategies such as TL or SSL.

Experiment	Metric	Rubble	Bulky items	Fire Wood	Scrap	Vehicles	Macro avg.
SDA (Uniform strategy + Blur)	*Precision*	60.00%	68.33%	50.00%	48.57%	100.00%	65.38%
	Recall	58.70%	85.42%	29.63%	60.71%	40.00%	54.89%
	F1-score	59.34%	**75.93%**	37.21%	53.97%	**57.14%**	**56.72%**
SDA + ImageNet Pre-training	*Precision*	58.14%	61.29%	47.37%	45.16%	100.00%	62.39%
	Recall	54.35%	79.17%	33.33%	50.00%	40.00%	51.37%
	F1-score	56.17%	69.09%	39.13%	47.46%	**57.14%**	53.80%
SDA + Image Rotations Pre-training	*Precision*	56.06%	61.54%	41.94%	42.31%	50.00%	50.37%
	Recall	80.43%	83.33%	48.15%	78.57%	20.00%	62.10%
	F1-score	**66.07%**	70.80%	**44.83%**	**55.00%**	28.57%	53.05%
SDA + Jigsaw puzzles Pre-training	*Precision*	65.71%	63.49%	41.67%	53.57%	100.00%	64.89%
	Recall	50.00%	83.33%	18.52%	53.57%	40.00%	49.08%
	F1-score	56.79%	72.07%	25.64%	53.57%	**57.14%**	53.04%
SDA + CAM-guided Tile2Vec Pre-training	*Precision*	56.82%	60.78%	66.67%	44.44%	100.00%	65.74%
	Recall	54.35%	64.58%	22.22%	14.29%	40.00%	39.09%
	F1-score	55.56%	62.63%	33.33%	21.62%	**57.14%**	46.06%

Multi-spectral Images Analysis

During the various experiments, a set of Near-Infrared (NIR) images for territories included in the AerialWaste data set was provided by ARPA Lombardy, the agency for environment protection. For this reason, a visual inspection of these images is conducted to verify whether the usage of other optical bands than RGB allows to better distinguish among the different types of landfills. Materials such as *glass, plexiglass, wood, brick, stone, asphalt* and *paper* all absorb Infrared radiation. Furthermore, different colors absorb radiation in different ways. Thus, it seems difficult that NIR can help in distinguishing better between the various classes. This is confirmed by the visual inspection, shown in Figure 4.9. The first row shows images containing a set of *Bulky items* and other small elements, which become indistinguishable in the NIR image. The bottom-left image shows an example in which *Rubble, Bulky items, Plastic, Domestic appliances* and *Paper* are present. However, once again, in the NIR image, these classes are not better distinguishable. Finally, the bottom-right image shows an example in which *Manure* is present, but in the NIR image it becomes similar to bales of hay.

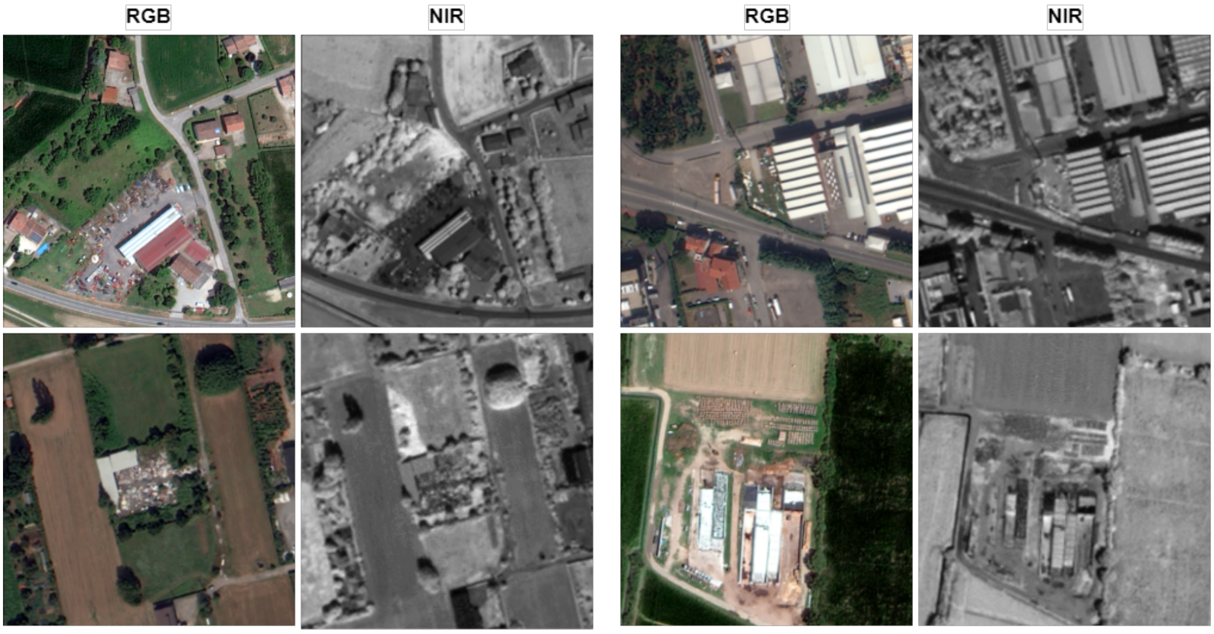


Figure 4.9: Examples of NIR images, compared to the original RGB images from the AerialWaste data set [18].

Evaluation on the Test Set

Given the previous analysis, the model based on Uniform Augmentation with Synthetic Data, using random positioning of the patches and contours blurring, pre-trained using Torres-AerialWaste weights is chosen for the evaluation on the test set and compared with the baseline to show the improvement. In this way, the generalization capabilities of the model can be assessed.

From a quantitative viewpoint, while the performance of the baseline degrades on the test set, the selected model shows similar results to the ones obtained on the validation set, as reported in 4.5. More specifically, while the average F1-score of the selected model is still around 56%, the performance of the baseline drops of 13%, demonstrating the importance of not choosing the model relying only on quantitative results. This shows that when a model seems able to effectively discriminate between classes, it is more prone to generalize better.

To complement the analysis, the CAMs generated by the best model are analyzed and visually compared to the segmentation masks available on some samples of the test set. In this way, a more clear idea of the discriminative capabilities of the model can be obtained. In general, the selected model is able to generate CAMs that are better than those generated by the baseline and at the same time, it is able to detect the correct classes in many cases, while the baseline often misleads a class for another. This

Table 4.5: Quantitative evaluation of the baseline and the best model on the test set.

Experiment	Metric	Rubble	Bulky items	Fire Wood	Scrap	Vehicles	Macro avg.
Baseline	*Precision*	70.00%	33.67%	53.13%	30.26%	70.00%	51.41%
	Recall	31.82%	75.00%	44.74%	85.19%	26.92%	52.73%
	F1-score	43.75%	46.48%	48.57%	44.66%	38.89%	44.47%
SDA (Uniform strategy + Blur)	*Precision*	69.49%	38.16%	48.72%	41.18%	92.86%	58.08%
	Recall	62.12%	65.91%	50.00%	77.78%	50.00%	61.16%
	F1-score	**65.60%**	**48.33%**	**49.35%**	**53.85%**	**65.00%**	**56.43%**

is displayed in Figure 4.10 which shows some examples of CAMs for a few test images, generated using the baseline and the selected model. The original image with the available segmentation masks is added for further comparison. The top image shows an example in which multiple instances of a single class (*Rubble*) are present. In this case, the baseline is not able to predict the presence of *Rubble* given that no attention is given to any of the present instances. Furthermore, it predicts the presence of both *Bulky items* and *Scrap*, which are not present even in the highlighted part. On the contrary, the selected model is not only able to correctly predict the presence of *Rubble* but also to focus on the whole more evident instance that is present. The other instances, instead, are much more difficult to detect, and, thus, no or little attention is given by the model to these instances. The second image shows another simple example in which multiple instances of *Scrap* are present. In this case, these instances are easier to find than those of the previous example. In fact, both the baseline and the selected model are able to correctly predict the presence of *Scrap*, focusing on the correct part of the image. Still, the CAM generated by the selected model is better than that of the baseline. Moreover, in this example, the baseline predicts also the presence of *Bulky items*, focusing on the exact same aspects as for the prediction of *Scrap*. This shows that the baseline is still not able to correctly distinguish among these classes. The bottom image shows an example of a sample in which only a couple of instances of class (*Rubble*) are present. In this case, the selected model correctly predicts the presence of *Rubble* and the CAM focuses mostly on the correct portion of the image. The second instance, however, is not detected by the selected model, probably due to the fact that only a portion of the instance is visible. Nonetheless, also in this case, the model is better than the baseline that, despite being able to correctly focus on the most visible instance, is unable to predict the presence of *Rubble*.

Figure 4.11 shows other examples in which the selected model performs well. More specifically, the top image shows an example in which there is only a single instance of *Fire Wood* in the image, which is difficult to detect even by visual inspection. Despite this, the model is able to correctly focus on the whole instance and predict the correct

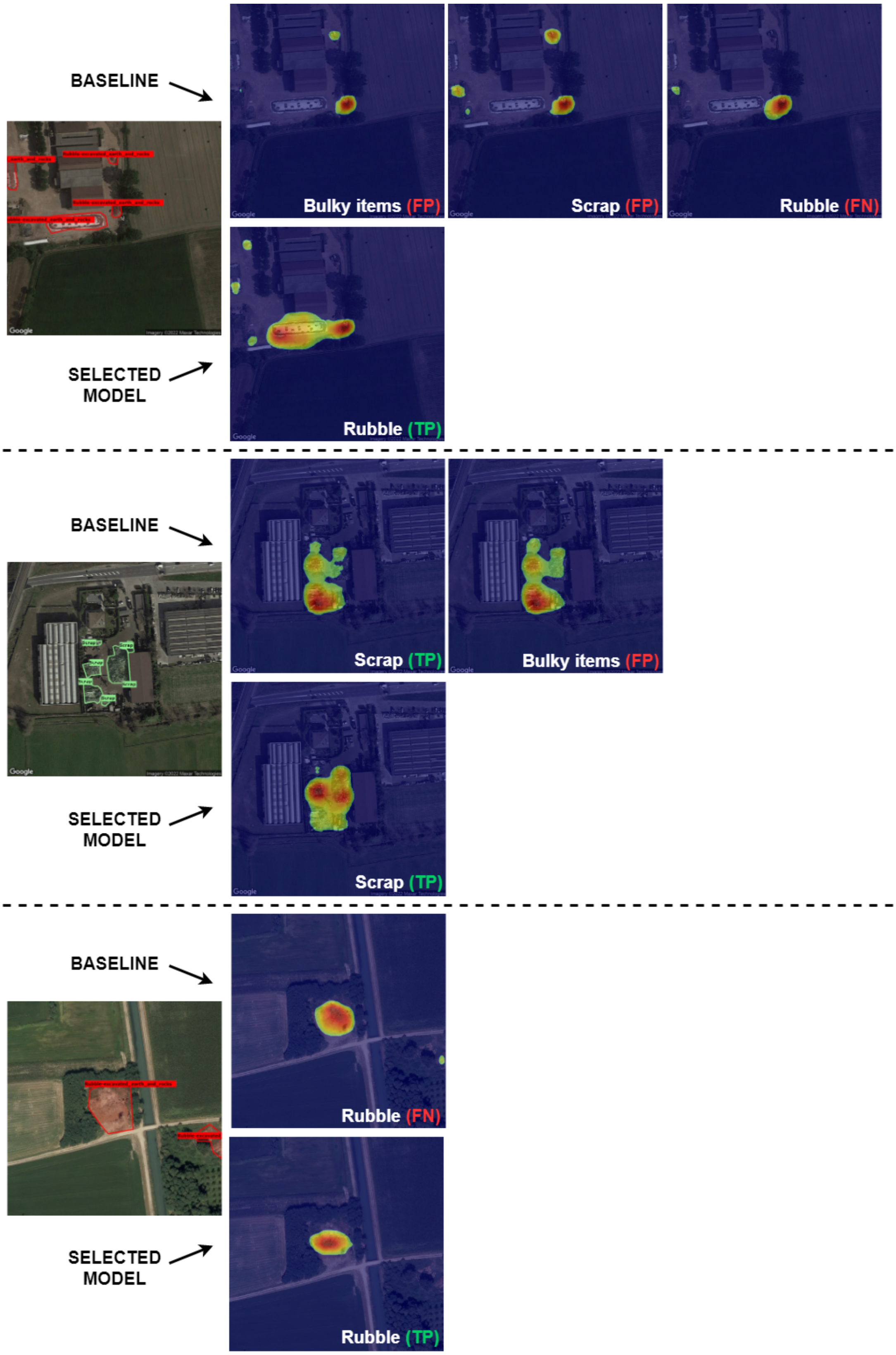


Figure 4.10: Examples of CAMs generated on test images. For each image, the top row shows CAMs generated by the baseline, whereas the bottom row shows CAMs generated by the selected model. It is clear that, in these cases, the selected model performs better than the baseline, making less confusion between classes.

class. The second image shows an example in which multiple instances of class *Vehicles* are present. In this case, however, there is another set of parked vehicles that are not abandoned, as indicated by the red bounding box. Surprisingly, the model is able to correctly distinguish between abandoned vehicles and parked ones. In fact, it correctly predicts the presence of *Vehicles* focusing only on those that are abandoned as shown in the CAM. The third image shows an example in which multiple instances of class *Fire Wood* are present. However, these instances are much different than that of the first image. In fact, while in the first case, a non-delimited heap of wood was present, in this image, wood is ordered in piles. Despite this, the model correctly detects a couple of instances, showing also the capability of the model to deal with the intra-class similarity. Finally, the last image shows an example in which both *Bulky items* and *Scrap* are present. In this case, the model correctly predicts the presence of both classes and focuses on different aspects. While the localization of *Scrap* is quite accurate, that of *Bulky items* is far from the GT segmentation masks. However, by inspecting the area highlighted by the model, it is possible to see that the objects on which the model is focusing can be considered similar to *Bulky items*. This is probably a missed instance, which is however correctly detected by the model.

Despite the previous results, the selected model still presents some limitations as displayed in Figure 4.12. The first image shows an example in which multiple instances of *Rubble* are present. In this case, both the baseline and the selected model correctly predict the presence of *Rubble*. However, while the instances cover most of the image, the CAMs are focused only on a small portion of the landfill. This can be due to the fact that in many cases, the illegal landfills cover a small portion of the image and only very few samples are present in which instances are big. Thus, the network is not able to detect very large instances when they are present. The second image shows an example in which a small instance of *Vehicles* is present. In this case, the selected model correctly focuses on the instance but is not able to predict the presence of the class. The same happens for the baseline which however predicts the presence of *Fire Wood* focusing on the vehicles, showing once again the fact that it is unable to differentiate between classes. The third image shows an example in which multiple instances of *Rubble* are present together with an instance of *Fire Wood*. In this case, the baseline correctly predicts the presence of *Rubble*, focusing on one of the instances. However, it is not able to predict *Fire Wood*, focusing on an instance of *Rubble*. The selected model instead is unable to detect both classes. Despite this, the CAM for class *Rubble* reveals that the network focused on an instance of *Rubble*, while the CAM for class *Fire Wood* is focused on the instance of *Fire Wood*. This shows that despite not being confident enough to predict the presence of the

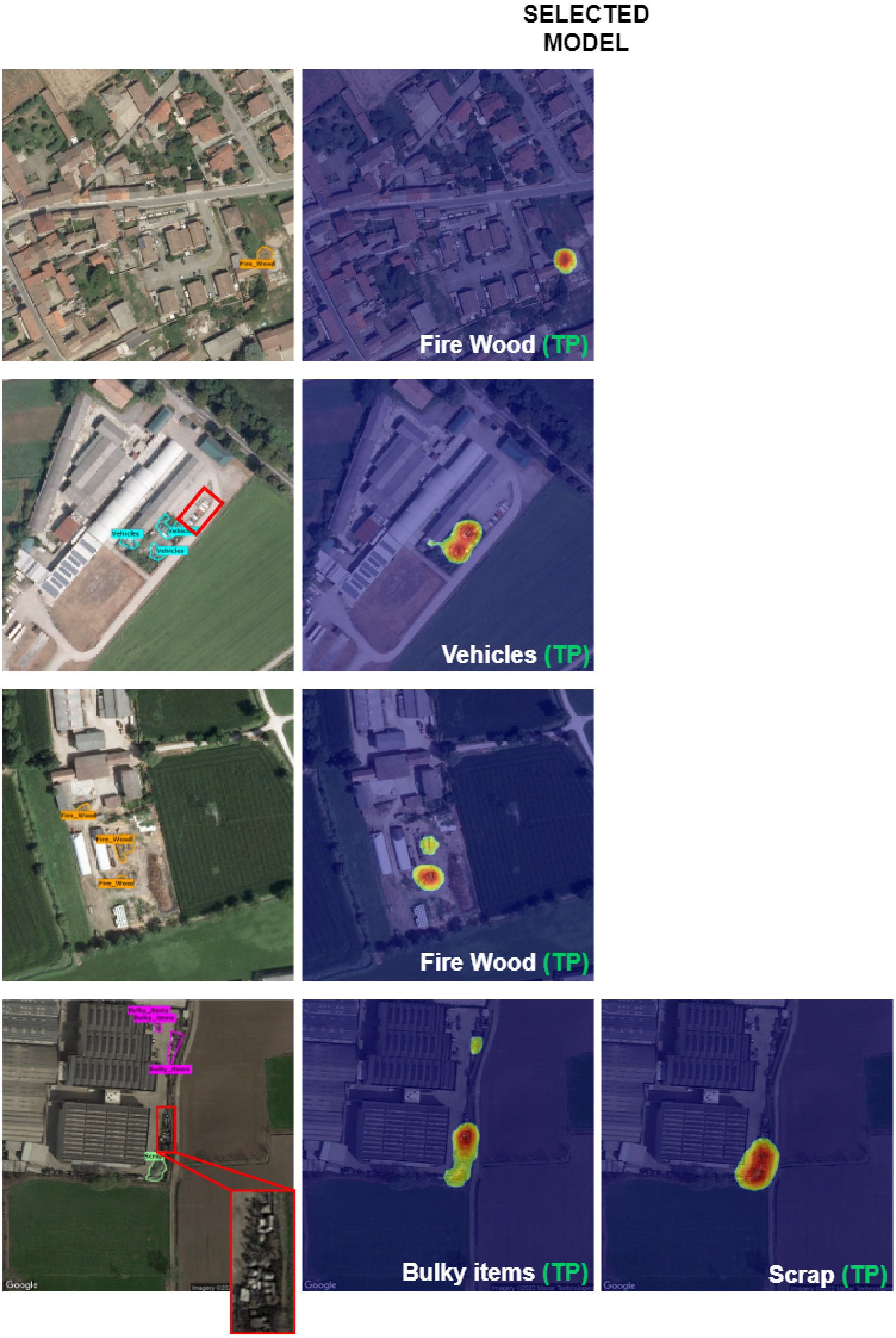


Figure 4.11: Examples of CAMs generated on test images by the selected model. In these cases, the selected model is able to correctly predict the classes that are present, focusing most of the time on the correct instances.

classes, the network looked at different aspects since it is more capable than the baseline to discriminate between the different classes. Finally, the last image shows an example containing an instance of *Bulky items*. In this case, both the models are not able to detect the instance probably due to the fact that shadows are present on the instance, making it less visible. Both models predict *Fire Wood* which is not present in the GT annotations. However, while the baseline focuses on something wrong, the selected model focuses on something that is clearly *Fire Wood*. Thus, once again, the chosen model is sometimes able to correctly detect types of landfills that are not annotated.

Given that the model presents some limitations since it is not able to detect large instances and sometimes even small ones, despite focusing on the right things, and still makes confusion when multiple classes are present, still, some work needs to be done. However, the above results show the superiority of the selected model with respect to the baseline from both a qualitative and quantitative standpoint, demonstrating the effectiveness of adding synthetic data, and in general, the importance of tackling the class co-occurrence problem.

Pseudo-label Generation

Considering the best classification model obtained so far, it is possible to evaluate the feasibility of a WSIS task. To this end, the model can be exploited to produce CAMs on the training images and then use these to generate pseudo-labels, as described in Section 3.3.

The direct segmentation of CAMs is performed using a binarization threshold of 0.5. The generated segmentations reflect all the limitations described during the qualitative evaluation of the CAMs. The detection and discrimination capabilities of the network are still limited due to the complexity of the task and the confusion between classes. For this reason, the segmentation masks are often overlapped, as shown in Figure 4.13, making it unfeasible to use them to train an instance segmentation network such as Mask R-CNN [52]. For instance, the second image shows an example in which the network correctly detects *Fire Wood* in the top-left corner. However, it also confuses the center part of the image which contains *Bulky items* (correctly identified) for *Fire Wood*. Thus, the generated segmentation masks are overlapped. The third image shows another example in which the obtained segmentation masks are of poor quality due to a bad output of the model. Using these pseudo-labels for Mask R-CNN would lead the network to a high level of confusion since the same instance would be fed to the network with multiple annotations.

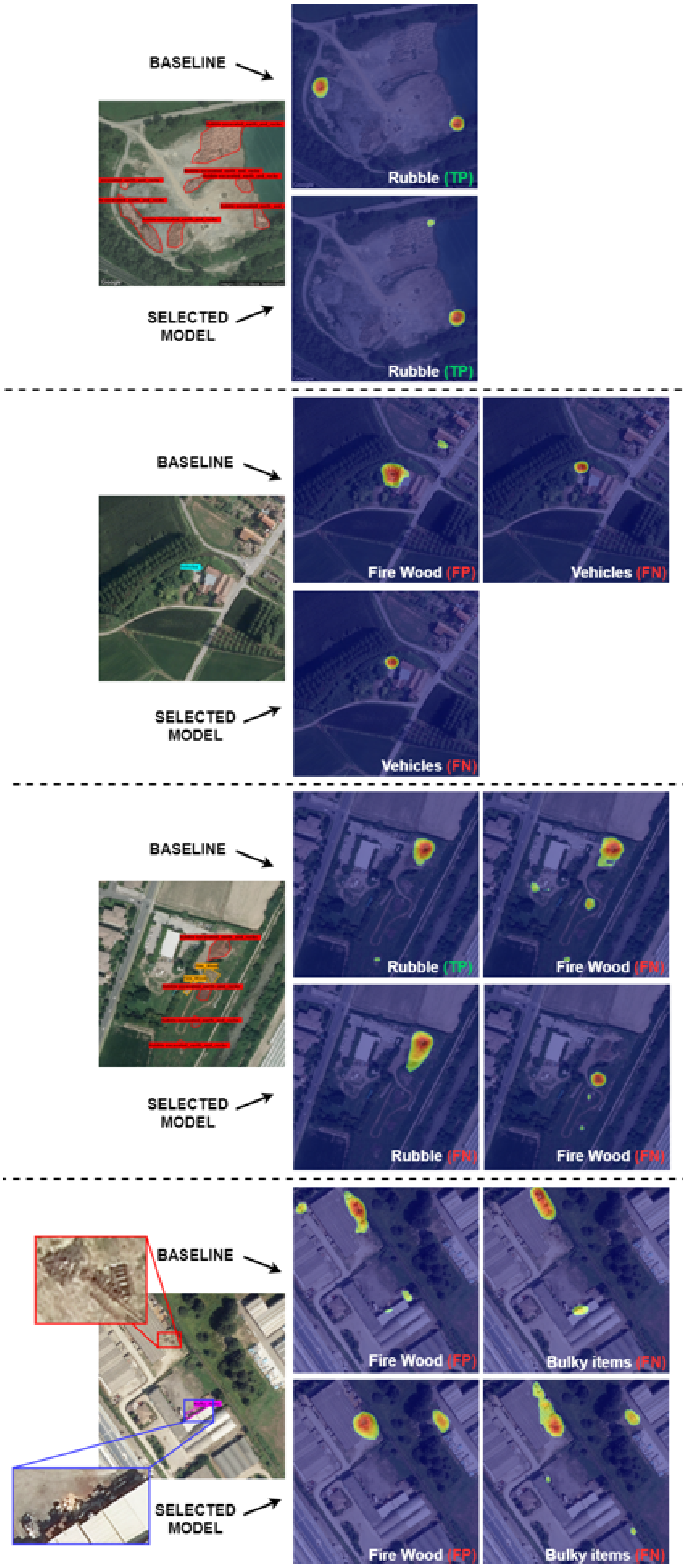


Figure 4.12: Examples of CAMs generated on test images. For each image, the top row shows CAMs generated by the baseline, whereas the bottom row shows CAMs generated by the selected model. In these cases, both models present some limitations.

Another experiment has been performed with IRNet [228] to obtain segmentation masks of higher quality. This approach exploits DenseCRF [288] to refine CAMs based on the borders of the instances. However, as shown in Figure 4.13, the results are even worse. This can be justified by the fact that the contours of the instances are not easily distinguishable from the background in RSIs and especially in the AerialWaste data set. In the first image, the segmentation obtained from the CAM is correct and quite good given that a simple example is considered. However, in this case, the segmentation obtained using IRNet has been extended too much. The results are even worse in the cases in which also the CAM-based approach is not good. Thus, regardless of the method used to generate segmentation masks, at the moment, the detection and discriminative capabilities reached by the multi-label classification model are not good enough to proceed with a WSIS task given that the quality of the generated pseudo-labels is very limited.

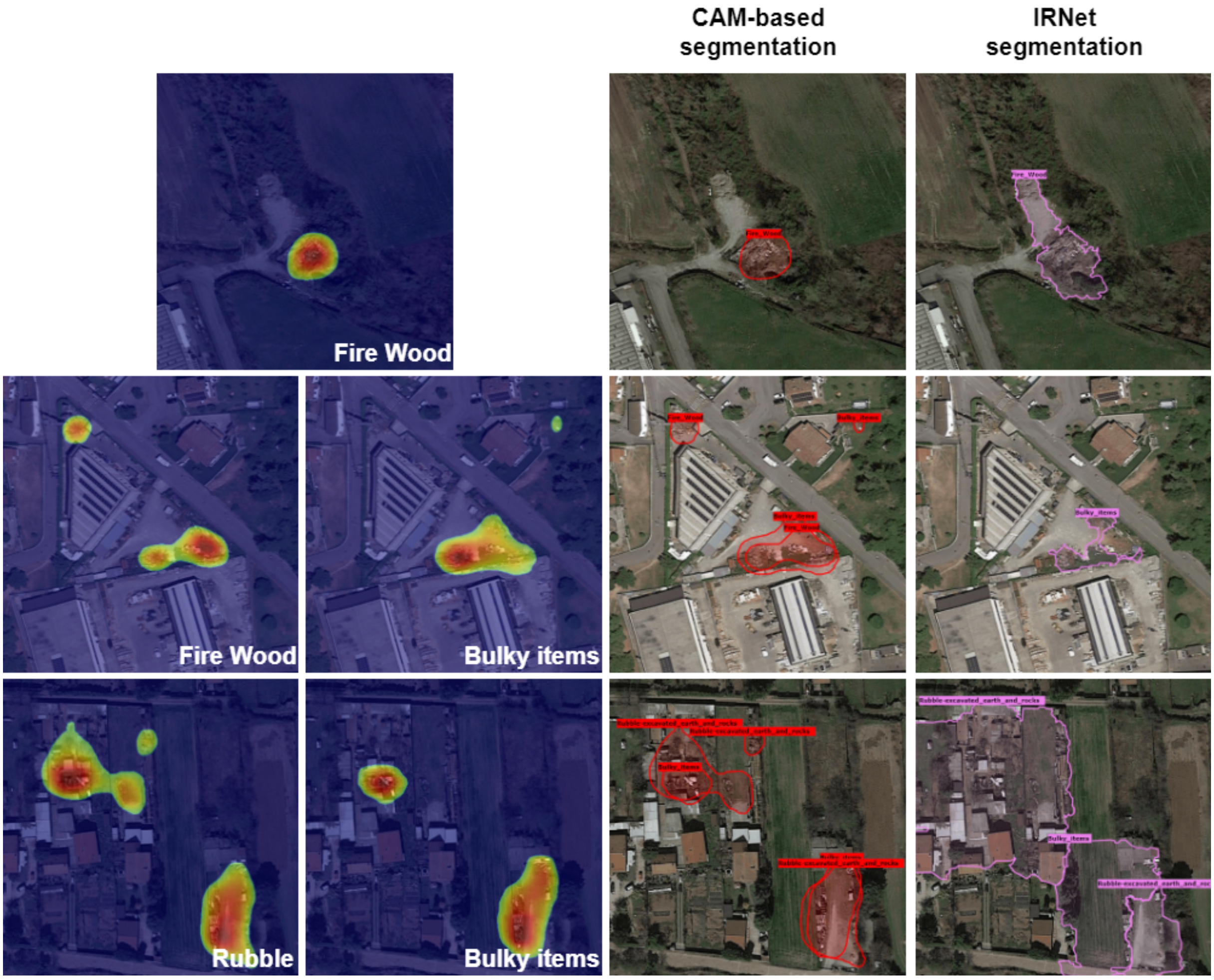


Figure 4.13: Examples of segmentation masks obtained from a direct segmentation of CAMs and using IRNet [228] for some training images. Those obtained directly from CAMs are generally better, but they still present major limitations.

5 | Conclusion and Future Works

In this thesis, the problem of Illegal Landfills detection was addressed as a multi-label classification problem. Particular emphasis was put on the discriminative and localization capabilities of the network to then design a weakly supervised localization task. A multi-label data set containing RSIs with illegal landfills was analyzed in depth, highlighting relevant characteristics that have a huge impact on classification performance, such as class co-occurrence and imbalance.

A ResNet50 backbone augmented with an FPN is used to improve the feature extraction at different scales as proposed by Torres et al. [8, 18]. The architecture was modified to allow the generation of the CAMs which are fundamental to perform a qualitative evaluation of the obtained models. Initially, preliminary single-class experiments are performed revealing the necessity of enlarging the training set and discarding classes that cannot be identified due to the very limited amount of samples.

Thus, an augmentation framework was designed to allow the extension of the training set with several strategies that try to mitigate the effects of class co-occurrence and imbalance. Among them, Synthetic Data Augmentation is the one that provides better results. An approach based on CAMs was further proposed to increase the realism of the synthetic samples, and a Linear Programming algorithm is designed with the aim of obtaining pre-defined desired characteristics in the final data set. However, the improvements provided by these two more sophisticated approaches are very limited, probably due to the complexity of the task.

Different initialization strategies were considered to initialize the backbone, revealing that using the weights provided by Torres et al. [18] was the most effective. To exploit the part of the data set not suitable for multi-label classification due to the availability of binary labels only, SSL approaches were analyzed and tested. A CAM-guided version of Tile2Vec was developed to learn better features of landfills. These approaches turned out to reach performances that are at most comparable with other experiments meaning that no more relevant features were learned.

A quantitative and qualitative evaluation of the models was performed using the

ODIN evaluation framework [19, 20].

The selected model was tested, achieving 56.43% average F1-score and satisfactory qualitative performances. The resulting model was considered to proceed with the WSL, and segmentation masks were generated starting from the CAMs generated by the model to be used as pseudo-label for the localization task, but their quality is not considered good enough to continue with an instance segmentation task. In order to further improve the results, future work will concentrate on:

- **Data set extension**: the analysis highlighted the fact that it is hard to further improve the result without increasing the size of the data set. In particular, new samples with multi-label annotations must be collected, paying attention to the co-occurrence of the classes inside them and the number of represented classes. Also, samples with coarse-grained annotation or no annotation could be useful to increase the performances, considering the possibility of pre-training the network with SSL.
- **Hyper-spectral Data**: the analysis performed on the images present in the data set showed a widespread inter-class similarity and intra-class diversity. However, these challenges are still present even if multi-spectral images are used. Considering other spectral bands that are usually present in hyper-spectral images may potentially alleviate this problem.
- **CAM-guided Tile2Vec**: the last part of the experiments shows an idea to extend the proposed SSL approach using the single-class classifiers to guide the extraction of the CAMS. This idea might be considered once the performances of the single-class classifiers are improved enough, for example, thanks to the data set extension. This approach could potentially lead to a valuable SSL strategy to exploit a large amount of unlabeled data.
- **RSI-specific WSL task**: given that the obtained model is considered not adequate to perform a WSIS task, sophisticated approaches for WSIS are not taken into account. However, once the desired detection capabilities are reached, more complex approaches such as those described in Chapter 2 can be used. If WSOD is considered instead of WSIS, Fasana et al. [13] provides a survey of RSWSOD methods that can be used as a starting point to build a proper RSWSOD model for illegal landfills detection.

Bibliography

[1] U. Desa *et al.*, “Transforming our world: The 2030 agenda for sustainable development,” 2016.

[2] L. S. Rame, A. Hartono, I. Firmansyah, *et al.*, “The effect of demographic factors on waste generation and heavy metal in illegal landfill at malaka regency, east nusa tenggara province,” in *IOP Conference Series: Earth and Environmental Science*, vol. 950, p. 012055, IOP Publishing, 2022.

[3] J. Szulc, M. Okrasa, A. Nowak, J. Nizioł, T. Ruman, and S. Kuberski, “Assessment of physicochemical, microbiological and toxicological hazards at an illegal landfill in central poland,” *International journal of environmental research and public health*, vol. 19, no. 8, p. 4826, 2022.

[4] A. Siddiqua, J. Hahladakis, and W. Al-Attiya, “An overview of the environmental pollution and health effects associated with waste landfilling and open dumping,” *Environmental Science and Pollution Research*, vol. 29, pp. 1–23, 08 2022.

[5] G. Wu, L. Wang, R. Yang, W. Hou, S. Zhang, X. Guo, and W. Zhao, “Pollution characteristics and risk assessment of heavy metals in the soil of a construction waste landfill site,” *Ecological Informatics*, p. 101700, 2022.

[6] *Ecomafia 2021. Le storie e i numeri della criminalità ambientale in Italia*. Edizioni Ambiente, 2021.

[7] G. E. Schrab, K. W. Brown, and K. Donnelly, “Acute and genetic toxicity of municipal landfill leachate,” *Water, Air, and Soil Pollution*, vol. 69, no. 1, pp. 99–112, 1993.

[8] R. N. Torres and P. Fraternali, “Learning to identify illegal landfills through scene classification in aerial images,” *Remote Sensing*, vol. 13, no. 22, p. 4520, 2021.

[9] Z. He, “Deep learning in image classification: A survey report,” in *2020 2nd International Conference on Information Technology and Computer Application (ITCA)*, pp. 174–177, 2020.

[10] M. Aljabri and M. AlGhamdi, "A review on the use of deep learning for medical images segmentation," *Neurocomputing*, 2022.

[11] M. Andriluka, J. R. R. Uijlings, and V. Ferrari, "Fluid annotation," in *Proceedings of the 26th ACM international conference on Multimedia*, ACM, oct 2018.

[12] F. Shao, L. Chen, J. Shao, W. Ji, S. Xiao, L. Ye, Y. Zhuang, and J. Xiao, "Deep learning for weakly-supervised object detection and localization: A survey," *Neurocomputing*, 2022.

[13] C. Fasana, S. Pasini, F. Milani, and P. Fraternali, "Weakly supervised object detection for remote sensing images: A survey," *Remote Sensing*, vol. 14, no. 21, 2022.

[14] S. Albelwi, "Survey on self-supervised learning: Auxiliary pretext tasks and contrastive learning methods in imaging," *Entropy*, vol. 24, p. 551, 04 2022.

[15] Y. Wang, C. M. Albrecht, N. A. A. Braham, L. Mou, and X. X. Zhu, "Self-supervised learning in remote sensing: A review," 2022.

[16] S. Abdukhamet, *Landfill Detection in Satellite Images Using Deep Learning.* PhD thesis, Shanghai Jiao Tong University Shanghai, China, 2019.

[17] O. Youme, T. Bayet, J. M. Dembele, and C. Cambier, "Deep learning and remote sensing: detection of dumping waste using uav," *Procedia Computer Science*, vol. 185, pp. 361–369, 2021.

[18] R. N. Torres and P. Fraternali, "Aerialwaste: A dataset for illegal landfill discovery in aerial images," Aug. 2022.

[19] R. N. Torres, P. Fraternali, and J. Romero, "Odin: an object detection and instance segmentation diagnosis framework," in *European Conference on Computer Vision*, pp. 19–31, Springer, 2020.

[20] R. N. Torres, F. Milani, and P. Fraternali, "Odin: Pluggable meta-annotations and metrics for the diagnosis of classification and localization," in *International Conference on Machine Learning, Optimization, and Data Science*, pp. 383–398, Springer, 2021.

[21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[22] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature

pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2117–2125, 2017.

[23] S. S. A. Zaidi, M. S. Ansari, A. Aslam, N. Kanwal, M. Asghar, and B. Lee, "A survey of modern deep learning based object detection models," *Digital Signal Processing*, p. 103514, 2022.

[24] A. M. Hafiz and G. M. Bhat, "A survey on instance segmentation: state of the art," *International Journal of Multimedia Information Retrieval*, vol. 9, pp. 171–189, jul 2020.

[25] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. van der Laak, B. van Ginneken, and C. I. Sánchez, "A survey on deep learning in medical image analysis," *Medical Image Analysis*, vol. 42, pp. 60–88, 2017.

[26] M. Zhang, Y. Zhou, J. Zhao, Y. Man, B. Liu, and R. Yao, "A survey of semi- and weakly supervised semantic segmentation of images," *Artificial Intelligence Review*, vol. 53, pp. 4259–4288, Aug 2020.

[27] Z. Zou, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," *arXiv preprint arXiv:1905.05055*, 2019.

[28] C. Sager, C. Janiesch, and P. Zschech, "A survey of image labelling for computer vision applications," *Journal of Business Analytics*, vol. 4, no. 2, pp. 91–110, 2021.

[29] R. N. Torres, "Analysis of geographic data for environmental monitoring," 2022.

[30] A. Krizhevsky, "Learning multiple layers of features from tiny images," *University of Toronto*, 05 2012.

[31] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

[32] M. Everingham, S. Eslami, L. Van Gool, C. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes challenge: A retrospective," *International Journal of Computer Vision*, vol. 111, 01 2014.

[33] T. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: common objects in context," *CoRR*, vol. abs/1405.0312, 2014.

[34] K. Li, G. Wan, G. Cheng, L. Meng, and J. Han, "Object detection in optical remote

sensing images: A survey and a new benchmark," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 159, pp. 296–307, jan 2020.

[35] G.-S. Xia, X. Bai, J. Ding, Z. Zhu, S. Belongie, J. Luo, M. Datcu, M. Pelillo, and L. Zhang, "Dota: A large-scale dataset for object detection in aerial images," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3974–3983, 2018.

[36] G. Cheng, J. Han, and X. Lu, "Remote sensing image scene classification: Benchmark and state of the art," *Proceedings of the IEEE*, vol. 105, no. 10, pp. 1865–1883, 2017.

[37] D. Zhang, J. Han, D. Yu, and J. Han, "Weakly supervised learning for airplane detection in remote sensing images," in *The Proceedings of the Second International Conference on Communications, Signal Processing, and Systems*, pp. 155–163, Springer, 2014.

[38] S. Wei, X. Zeng, Q. Qu, M. Wang, H. Su, and J. Shi, "Hrsid: A high-resolution sar images dataset for ship detection and instance segmentation," *IEEE Access*, vol. 8, pp. 120234–120254, 2020.

[39] Y. Hu, Y. Li, and Z. Pan, "A dual-polarimetric sar ship detection dataset and a memory-augmented autoencoder-based detection method," *Sensors*, vol. 21, no. 24, p. 8478, 2021.

[40] Y. Yang, Z. Pan, Y. Hu, and C. Ding, "Pistonnet: Object separating from background by attention for weakly supervised ship detection," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 15, pp. 5190–5202, 2022.

[41] M. Cramer, "The dgpf-test on digital airborne camera evaluation overview and test design," *Photogrammetrie - Fernerkundung - Geoinformation*, vol. 2010, pp. 73–82, 05 2010.

[42] D. Zhang, J. Han, G. Cheng, Z. Liu, S. Bu, and L. Guo, "Weakly supervised learning for target detection in remote sensing images," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 4, pp. 701–705, 2014.

[43] G. Cheng, J. Han, P. Zhou, and L. Guo, "Multi-class geospatial object detection and geographic image classification based on collection of part detectors," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 98, pp. 119–132, 2014.

[44] K. Li, G. Cheng, S. Bu, and X. You, "Rotation-insensitive and context-augmented

object detection in remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 4, pp. 2337–2348, 2017.

[45] Z. Zou and Z. Shi, "Random access memories: A new paradigm for target detection in high resolution aerial remote sensing images," *IEEE Transactions on Image Processing*, vol. 27, no. 3, pp. 1100–1111, 2018.

[46] Y. Zhang, Y. Yuan, Y. Feng, and X. Lu, "Hierarchical and robust convolutional neural network for very high-resolution remote sensing object detection," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 8, pp. 5535–5548, 2019.

[47] Z.-Z. Wu, T. Weise, Y. Wang, and Y. Wang, "Convolutional neural network based weakly supervised learning for aircraft detection from remote sensing image," *IEEE Access*, vol. 8, pp. 158097–158106, 2020.

[48] H. Chen and Z. Shi, "A spatial-temporal attention-based method and a new dataset for remote sensing image change detection," *Remote Sensing*, vol. 12, no. 10, 2020.

[49] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[50] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[51] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," 2015.

[52] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[53] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[54] N. Cristianini and E. Ricci, *Support Vector Machines*, pp. 928–932. Boston, MA: Springer US, 2008.

[55] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014.

[56] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg,

"SSD: Single shot MultiBox detector," in *Computer Vision – ECCV 2016*, pp. 21–37, Springer International Publishing, 2016.

[57] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," 2015.

[58] A. Bearman, O. Russakovsky, V. Ferrari, and L. Fei-Fei, "What's the point: Semantic segmentation with point supervision," 2015.

[59] X. Yu, J. Wang, Q.-Q. Hong, R. Teku, S.-H. Wang, and Y.-D. Zhang, "Transfer learning for medical images analyses: A survey," *Neurocomputing*, vol. 489, pp. 230–254, 2022.

[60] Y. Chen, M. Mancini, X. Zhu, and Z. Akata, "Semi-supervised and unsupervised deep visual learning: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–23, 2022.

[61] J. Yue, L. Fang, P. Ghamisi, W. Xie, J. Li, J. Chanussot, and A. Plaza, "Optical remote sensing image understanding with weak supervision: Concepts, methods, and perspectives," *IEEE Geoscience and Remote Sensing Magazine*, vol. 10, no. 2, pp. 250–269, 2022.

[62] M. Lu, L. Fang, M. Li, B. Zhang, Y. Zhang, and P. Ghamisi, "Nfanet: A novel method for weakly supervised water extraction from high-resolution remote-sensing imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–14, 2022.

[63] Y. Li, W. Chen, Y. Zhang, C. Tao, R. Xiao, and Y. Tan, "Accurate cloud detection in high-resolution remote sensing imagery by weakly supervised deep learning," *Remote Sensing of Environment*, vol. 250, p. 112045, 2020.

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

[65] X. Qian, Y. Huo, G. Cheng, X. Yao, K. Li, H. Ren, and W. Wang, "Incorporating the completeness and difficulty of proposals into weakly supervised object detection in remote sensing images," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 15, pp. 1902–1911, 2022.

[66] L. Chen, Y. Fu, S. You, and H. Liu, "Efficient hybrid supervision for instance segmentation in aerial images," *Remote. Sens.*, vol. 13, p. 252, 2021.

[67] H. Wang, H. Li, W. Qian, W. Diao, L. Zhao, J. Zhang, and D. Zhang, "Dynamic pseudo-label generation for weakly supervised object detection in remote sensing images," *Remote Sensing*, vol. 13, no. 8, p. 1461, 2021.

[68] R. Bro and A. K. Smilde, "Principal component analysis," *Analytical methods*, vol. 6, no. 9, pp. 2812–2831, 2014.

[69] W. Jia, M. Sun, J. Lian, and S. Hou, "Feature dimensionality reduction: a review," *Complex & Intelligent Systems*, pp. 1–31, 2022.

[70] A. E. Ezugwu, A. M. Ikotun, O. O. Oyelade, L. Abualigah, J. O. Agushaka, C. I. Eke, and A. A. Akinyelu, "A comprehensive survey of clustering algorithms: State-of-the-art machine learning applications, taxonomy, challenges, and future research prospects," *Engineering Applications of Artificial Intelligence*, vol. 110, p. 104743, 2022.

[71] P. Goyal, M. Caron, B. Lefaudeux, M. Xu, P. Wang, V. Pai, M. Singh, V. Liptchinsky, I. Misra, A. Joulin, *et al.*, "Self-supervised pretraining of visual features in the wild," *arXiv preprint arXiv:2103.01988*, 2021.

[72] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th international conference on Machine learning*, pp. 1096–1103, 2008.

[73] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16000–16009, 2022.

[74] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems* (Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, eds.), vol. 27, Curran Associates, Inc., 2014.

[75] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[76] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," *arXiv preprint arXiv:1803.07728*, 2018.

[77] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," in *European conference on computer vision*, pp. 649–666, Springer, 2016.

[78] I. Misra, C. L. Zitnick, and M. Hebert, "Shuffle and learn: unsupervised learn-

ing using temporal order verification," in *European conference on computer vision*, pp. 527–544, Springer, 2016.

[79] C. Doersch, A. Gupta, and A. A. Efros, "Unsupervised visual representation learning by context prediction," in *Proceedings of the IEEE international conference on computer vision*, pp. 1422–1430, 2015.

[80] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a" siamese" time delay neural network," *Advances in neural information processing systems*, vol. 6, 1993.

[81] L. Jing, P. Vincent, Y. LeCun, and Y. Tian, "Understanding dimensional collapse in contrastive self-supervised learning," *arXiv preprint arXiv:2110.09348*, 2021.

[82] I. Misra and L. v. d. Maaten, "Self-supervised learning of pretext-invariant representations," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6707–6717, 2020.

[83] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9729–9738, 2020.

[84] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*, pp. 1597–1607, PMLR, 2020.

[85] L. Xu, J. Lian, W. X. Zhao, M. Gong, L. Shou, D. Jiang, X. Xie, and J.-R. Wen, "Negative sampling for contrastive representation learning: A review," *arXiv preprint arXiv:2206.00212*, 2022.

[86] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, "Deep clustering for unsupervised learning of visual features," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 132–149, 2018.

[87] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised learning of visual features by contrasting cluster assignments," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9912–9924, 2020.

[88] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. Richemond, E. Buchatskaya, C. Doersch, B. Avila Pires, Z. Guo, M. Gheshlaghi Azar, *et al.*, "Bootstrap your own latent-a new approach to self-supervised learning," *Advances in neural information processing systems*, vol. 33, pp. 21271–21284, 2020.

[89] G. Hinton, O. Vinyals, J. Dean, *et al.*, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, vol. 2, no. 7, 2015.

[90] J. Zbontar, L. Jing, I. Misra, Y. LeCun, and S. Deny, "Barlow twins: Self-supervised learning via redundancy reduction," in *International Conference on Machine Learning*, pp. 12310–12320, PMLR, 2021.

[91] H. B. Barlow *et al.*, "Possible principles underlying the transformation of sensory messages," *Sensory communication*, vol. 1, no. 01, 1961.

[92] D. Bank, N. Koenigstein, and R. Giryes, "Autoencoders," 2020.

[93] J. Donahue, P. Krähenbühl, and T. Darrell, "Adversarial feature learning," *arXiv preprint arXiv:1605.09782*, 2016.

[94] A. Ng *et al.*, "Sparse autoencoder," *CS294A Lecture notes*, vol. 72, no. 2011, pp. 1–19, 2011.

[95] R. Salah, P. Vincent, X. Muller, *et al.*, "Contractive auto-encoders: Explicit invariance during feature extraction," in *Proc. of the 28th International Conference on Machine Learning*, pp. 833–840, 2011.

[96] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2536–2544, 2016.

[97] R. Zhang, P. Isola, and A. A. Efros, "Split-brain autoencoders: Unsupervised learning by cross-channel prediction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[98] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," 2020.

[99] M. Noroozi and P. Favaro, "Unsupervised learning of visual representations by solving jigsaw puzzles," in *European conference on computer vision*, pp. 69–84, Springer, 2016.

[100] R. Santa Cruz, B. Fernando, A. Cherian, and S. Gould, "Visual permutation learning," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 12, pp. 3100–3114, 2018.

[101] P. Chen, S. Liu, and J. Jia, "Jigsaw clustering for unsupervised visual representation

learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11526–11535, 2021.

[102] M. Noroozi, A. Vinjimoor, P. Favaro, and H. Pirsiavash, "Boosting self-supervised learning via knowledge transfer," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 9359–9367, 2018.

[103] M. Noroozi, H. Pirsiavash, and P. Favaro, "Representation learning by learning to count," in *Proceedings of the IEEE international conference on computer vision*, pp. 5898–5906, 2017.

[104] G. Larsson, M. Maire, and G. Shakhnarovich, "Colorization as a proxy task for visual understanding," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6874–6883, 2017.

[105] D. Kim, D. Cho, and I. S. Kweon, "Self-supervised video representation learning with space-time cubic puzzles," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, pp. 8545–8552, 2019.

[106] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance discrimination," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3733–3742, 2018.

[107] J. A. Hartigan, *Clustering Algorithms.* USA: John Wiley & Sons, Inc., 99th ed., 1975.

[108] G. Hinton, O. Vinyals, J. Dean, *et al.*, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, vol. 2, no. 7, 2015.

[109] X. Chen and K. He, "Exploring simple siamese representation learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15750–15758, 2021.

[110] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging properties in self-supervised vision transformers," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9650–9660, 2021.

[111] C. Li, J. Yang, P. Zhang, M. Gao, B. Xiao, X. Dai, L. Yuan, and J. Gao, "Efficient self-supervised vision transformers for representation learning," *arXiv preprint arXiv:2106.09785*, 2021.

[112] P. Zhang, M. Gong, L. Su, J. Liu, and Z. Li, "Change detection based on deep feature representation and mapping transformation for multi-spatial-resolution remote

sensing images," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 116, pp. 24–41, 2016.

[113] X. Lu, X. Zheng, and Y. Yuan, "Remote sensing scene classification by unsupervised representation learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 9, pp. 5148–5157, 2017.

[114] L. Mou, P. Ghamisi, and X. X. Zhu, "Unsupervised spectral–spatial feature learning via deep residual conv–deconv network for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 1, pp. 391–406, 2017.

[115] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 2, pp. 2169–2178, 2006.

[116] L. Zhu, Y. Chen, P. Ghamisi, and J. A. Benediktsson, "Generative adversarial networks for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 9, pp. 5046–5063, 2018.

[117] Q. Jin, Y. Ma, F. Fan, J. Huang, X. Mei, and J. Ma, "Adversarial autoencoder network for hyperspectral unmixing," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[118] L. H. Hughes, M. Schmitt, and X. X. Zhu, "Mining hard negative samples for sar-optical image matching using generative adversarial networks," *Remote Sensing*, vol. 10, no. 10, p. 1552, 2018.

[119] J. L. H. Alvarez, M. Ravanbakhsh, and B. Demir, "S2-cgan: Self-supervised adversarial representation learning for binary change detection in multispectral images," in *IGARSS 2020-2020 IEEE International Geoscience and Remote Sensing Symposium*, pp. 2515–2518, IEEE, 2020.

[120] K. Walter, M. J. Gibson, and A. Sowmya, "Self-supervised remote sensing image retrieval," in *IGARSS 2020-2020 IEEE International Geoscience and Remote Sensing Symposium*, pp. 1683–1686, IEEE, 2020.

[121] G. Cheng, X. Sun, K. Li, L. Guo, and J. Han, "Perturbation-seeking generative adversarial networks: A defense framework for remote sensing image scene classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–11, 2021.

[122] S. Ozkan and G. B. Akar, "Spectral unmixing with multinomial mixture kernel and wasserstein generative adversarial loss," *arXiv preprint arXiv:2012.06859*, 2020.

[123] S. Singh, A. Batra, G. Pang, L. Torresani, S. Basu, M. Paluri, and C. Jawahar, "Self-supervised feature learning for semantic segmentation of overhead imagery.," in *BMVC*, vol. 1, p. 4, 2018.

[124] S. Zhang, Z. Wen, Z. Liu, and Q. Pan, "Rotation awareness based self-supervised learning for sar target recognition," in *IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium*, pp. 1378–1381, IEEE, 2019.

[125] C. Tao, J. Qi, W. Lu, H. Wang, and H. Li, "Remote sensing image scene classification with self-supervised paradigm under limited labeled samples," *IEEE Geoscience and Remote Sensing Letters*, 2020.

[126] Z. Zhao, Z. Luo, J. Li, C. Chen, and Y. Piao, "When self-supervised learning meets scene classification: Remote sensing scene classification based on a multitask learning framework," *Remote Sensing*, vol. 12, no. 20, p. 3276, 2020.

[127] H. Ji, Z. Gao, Y. Zhang, Y. Wan, C. Li, and T. Mei, "Few-shot scene classification of optical remote sensing images leveraging calibrated pretext tasks," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–13, 2022.

[128] S. Vincenzi, A. Porrello, P. Buzzega, M. Cipriano, P. Fronte, R. Cuccu, C. Ippoliti, A. Conte, and S. Calderara, "The color out of space: learning self-supervised representations for earth observation imagery," in *2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 3034–3041, IEEE, 2021.

[129] X. Wu, D. Hong, and D. Zhao, "Hyper-embedder: Learning a deep embedder for self-supervised hyperspectral dimensionality reduction," *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1–5, 2021.

[130] H. Dong, W. Ma, Y. Wu, J. Zhang, and L. Jiao, "Self-supervised representation learning for remote sensing image change detection based on temporal prediction," *Remote Sensing*, vol. 12, no. 11, p. 1868, 2020.

[131] Y. Yuan and L. Lin, "Self-supervised pretraining of transformers for satellite image time series classification," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 474–487, 2020.

[132] Y. Yuan, L. Lin, Q. Liu, R. Hang, and Z.-G. Zhou, "Sits-former: A pre-trained spatio-spectral-temporal representation model for sentinel-2 time series classifica-

tion," *International Journal of Applied Earth Observation and Geoinformation*, vol. 106, p. 102651, 2022.

[133] N. Jean, S. Wang, A. Samar, G. Azzari, D. Lobell, and S. Ermon, "Tile2vec: Unsupervised representation learning for spatially distributed data," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 3967–3974, 2019.

[134] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[135] H. Jung and T. Jeon, "Self-supervised learning with randomised layers for remote sensing," *Electronics Letters*, vol. 57, no. 6, pp. 249–251, 2021.

[136] M. Leenstra, D. Marcos, F. Bovolo, and D. Tuia, "Self-supervised pre-training enhances change detection in sentinel-2 imagery," in *International Conference on Pattern Recognition*, pp. 578–590, Springer, 2021.

[137] S. Hou, H. Shi, X. Cao, X. Zhang, and L. Jiao, "Hyperspectral imagery classification based on contrastive learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–13, 2021.

[138] M. Zhu, J. Fan, Q. Yang, and T. Chen, "Sc-eadnet: A self-supervised contrastive efficient asymmetric dilated network for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–17, 2021.

[139] L. Zhao, W. Luo, Q. Liao, S. Chen, and J. Wu, "Hyperspectral image classification with contrastive self-supervised learning under limited labeled samples," *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1–5, 2022.

[140] J. Kang, R. Fernandez-Beltran, P. Duan, S. Liu, and A. J. Plaza, "Deep unsupervised embedding for remotely sensed images based on spatially augmented momentum contrast," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 3, pp. 2598–2610, 2020.

[141] H. Jung, Y. Oh, S. Jeong, C. Lee, and T. Jeon, "Contrastive self-supervised learning with smoothed representation for remote sensing," *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1–5, 2021.

[142] H. Li, Y. Li, G. Zhang, R. Liu, H. Huang, Q. Zhu, and C. Tao, "Remote sensing images semantic segmentation with general remote sensing vision model via a self-supervised contrastive learning method," *arXiv preprint arXiv:2106.10605*, 2021.

[143] A. Montanaro, D. Valsesia, G. Fracastoro, and E. Magli, "Semi-supervised learn-

ing for joint sar and multispectral land cover classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1–5, 2022.

[144] L. Scheibenreif, J. Hanna, M. Mommert, and D. Borth, "Self-supervised vision transformers for land-cover segmentation and classification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1422–1431, 2022.

[145] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10012–10022, 2021.

[146] L. Scheibenreif, M. Mommert, and D. Borth, "Contrastive self-supervised data fusion for satellite imagery," *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. V-3-2022, pp. 705–711, 05 2022.

[147] M. Stevenson, C. Mues, and C. Bravo, "Deep residential representations: Using unsupervised learning to unlock elevation data for geo-demographic prediction," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 187, pp. 378–392, 2022.

[148] V. Stojnic and V. Risojevic, "Self-supervised learning of remote sensing scene representations using contrastive multiview coding," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1182–1191, 2021.

[149] Y. Chen and L. Bruzzone, "Self-supervised remote sensing images change detection at pixel-level," *arXiv preprint arXiv:2105.08501*, 2021.

[150] B. Liu, A. Yu, X. Yu, R. Wang, K. Gao, and W. Guo, "Deep multiview learning for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 9, pp. 7758–7772, 2020.

[151] K. Ayush, B. Uzkent, C. Meng, K. Tanmay, M. Burke, D. Lobell, and S. Ermon, "Geography-aware self-supervised learning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10181–10190, 2021.

[152] K. Heidler, L. Mou, D. Hu, P. Jin, G. Li, C. Gan, J.-R. Wen, and X. X. Zhu, "Self-supervised audiovisual representation learning for remote sensing data," *arXiv preprint arXiv:2108.00688*, 2021.

[153] S. Saha, P. Ebel, and X. X. Zhu, "Self-supervised multisensor change detection," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–10, 2021.

[154] X. Chen, H. Fan, R. Girshick, and K. He, “Improved baselines with momentum contrastive learning,” *arXiv preprint arXiv:2003.04297*, 2020.

[155] S. Saha, M. Shahzad, L. Mou, Q. Song, and X. X. Zhu, “Unsupervised single-scene semantic segmentation for earth observation,” *IEEE Transactions on Geoscience and Remote Sensing*, 2022.

[156] F. Liu, X. Qian, L. Jiao, X. Zhang, L. Li, and Y. Cui, “Contrastive learning-based dual dynamic gcn for sar image scene classification,” *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[157] D. Guo, Y. Xia, and X. Luo, “Self-supervised gans with similarity loss for remote sensing image scene classification,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 2508–2521, 2021.

[158] X. Hu, T. Li, T. Zhou, Y. Liu, and Y. Peng, “Contrastive learning based on transformer for hyperspectral image classification,” *Applied Sciences*, vol. 11, no. 18, p. 8670, 2021.

[159] D. Muhtar, X. Zhang, and P. Xiao, “Index your position: A novel self-supervised learning method for remote sensing images semantic segmentation,” *IEEE Transactions on Geoscience and Remote Sensing*, 2022.

[160] H. Chen, W. Li, S. Chen, and Z. Shi, “Semantic-aware dense representation learning for remote sensing image change detection,” *arXiv preprint arXiv:2205.13769*, 2022.

[161] X. Zhang, L. Han, T. Sobeih, L. Lappin, M. A. Lee, A. Howard, and A. Kisdi, “The self-supervised spectral–spatial vision transformer network for accurate prediction of wheat nitrogen status from uav imagery,” *Remote Sensing*, vol. 14, no. 6, p. 1400, 2022.

[162] L. Zhang, J. Song, A. Gao, J. Chen, C. Bao, and K. Ma, “Be your own teacher: Improve the performance of convolutional neural networks via self distillation,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3713–3722, 2019.

[163] V. Marsocci and S. Scardapane, “Continual barlow twins: continual self-supervised learning for remote sensing semantic segmentation,” *arXiv preprint arXiv:2205.11319*, 2022.

[164] N. A. A. Braham, L. Mou, J. Chanussot, J. Mairal, and X. X. Zhu, “Self supervised learning for few shot hyperspectral image classification,” in *IGARSS 2022-*

2022 IEEE International Geoscience and Remote Sensing Symposium, pp. 267–270, IEEE, 2022.

[165] L. Cao, F. Luo, L. Chen, Y. Sheng, H. Wang, C. Wang, and R. Ji, "Weakly supervised vehicle detection in satellite images via multi-instance discriminative learning," *Pattern Recognition*, vol. 64, pp. 417–424, 2017.

[166] Y. Sheng, L. Cao, C. Wang, and J. Li, "Weakly supervised vehicle detection in satellite images via multiple instance ranking," in *2018 24th International Conference on Pattern Recognition (ICPR)*, pp. 2765–2770, IEEE, 2018.

[167] S. Srivastava, S. Narayan, and S. Mittal, "A survey of deep learning techniques for vehicle detection from uav images," *Journal of Systems Architecture*, vol. 117, p. 102152, 2021.

[168] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders, "Selective search for object recognition," *International Journal of Computer Vision*, vol. 104, pp. 154–171, Sep 2013.

[169] C. L. Zitnick and P. Dollár, "Edge boxes: Locating object proposals from edges," in *Computer Vision – ECCV 2014* (D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, eds.), (Cham), pp. 391–405, Springer International Publishing, 2014.

[170] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, pp. 2980–2988, 2017.

[171] J. Pont-Tuset, P. Arbelaez, J. T.Barron, F. Marques, and J. Malik, "Multiscale combinatorial grouping for image segmentation and object proposal generation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, pp. 128–140, jan 2017.

[172] Y. Zhou, Y. Zhu, Q. Ye, Q. Qiu, and J. Jiao, "Weakly supervised instance segmentation using class peak response," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3791–3800, 2018.

[173] A. Arun, C. V. Jawahar, and M. P. Kumar, "Weakly supervised instance segmentation by learning annotation consistent instances," in *Computer Vision – ECCV 2020* (A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, eds.), (Cham), pp. 254–270, Springer International Publishing, 2020.

[174] W. He, X.-Y. Zhang, F. Yin, Z. Luo, J.-M. Ogier, and C.-L. Liu, "Realtime multi-

scale scene text detection with scale-based region proposal network," *Pattern Recognition*, vol. 98, p. 107026, 2020.

[175] Z.-Z. Wu, J. Xu, Y. Wang, F. Sun, M. Tan, and T. Weise, "Hierarchical fusion and divergent activation based weakly supervised learning for object detection from remote sensing images," *Information Fusion*, vol. 80, pp. 23–43, 2022.

[176] Y. Li, B. He, F. Melgani, and T. Long, "Point-based weakly supervised learning for object detection in high spatial resolution remote sensing images," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 5361–5371, 2021.

[177] L. Chen, T. Yang, X. Zhang, W. Zhang, and J. Sun, "Points as queries: Weakly semi-supervised object detection by points," 2021.

[178] H. Bilen and A. Vedaldi, "Weakly supervised deep detection networks," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2846–2854, 2016.

[179] P. Tang, X. Wang, X. Bai, and W. Liu, "Multiple instance detection network with online instance classifier refinement," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3059–3067, 2017.

[180] P. Tang, X. Wang, S. Bai, W. Shen, X. Bai, W. Liu, and A. Yuille, "Pcl: Proposal cluster learning for weakly supervised object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, pp. 1–1, 10 2018.

[181] Z. Chen, Z. Fu, R. Jiang, Y. Chen, and X.-s. Hua, "Slv: Spatial likelihood voting for weakly supervised object detection," 2020.

[182] G. Cheng, J. Yang, D. Gao, L. Guo, and J. Han, "High-quality proposals for weakly supervised object detection," *IEEE Transactions on Image Processing*, vol. 29, pp. 5794–5804, 2020.

[183] N. Gonthier, S. Ladjal, and Y. Gousseau, "Multiple instance learning on deep features for weakly supervised object detection with extreme domain shifts," *Computer Vision and Image Understanding*, vol. 214, p. 103299, 2022.

[184] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez, "Solving the multiple instance problem with axis-parallel rectangles," *Artificial Intelligence*, vol. 89, no. 1, pp. 31–71, 1997.

[185] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Object detectors emerge in deep scene cnns," 2014.

[186] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Is object localization for free? - weakly-supervised learning with convolutional neural networks," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 685–694, 2015.

[187] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2921–2929, 2016.

[188] S. Yang, Y. Kim, Y. Kim, and C. Kim, "Combinational class activation maps for weakly supervised object localization," 2019.

[189] H. Xue, C. Liu, F. Wan, J. Jiao, X. Ji, and Q. Ye, "Danet: Divergent activation for weakly supervised object localization," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6588–6597, 2019.

[190] J. Mai, M. Yang, and W. Luo, "Erasing integrated learning: A simple yet effective approach for weakly supervised object localization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[191] J. Wei, Q. Wang, Z. Li, S. Wang, S. K. Zhou, and S. Cui, "Shallow feature matters for weakly supervised object localization," 2021.

[192] K. Wang, J. Oramas, and T. Tuytelaars, "Minmaxcam: Improving object coverage for cam-basedweakly supervised object localization," 2021.

[193] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 618–626, 2017.

[194] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," in *Computer Vision – ECCV 2014*, pp. 346–361, Springer International Publishing, 2014.

[195] X. Li, M. Kan, S. Shan, and X. Chen, "Weakly supervised object detection with segmentation collaboration," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[196] X. Feng, J. Han, X. Yao, and G. Cheng, "Progressive contextual instance refinement

for weakly supervised object detection in remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 11, pp. 8002–8012, 2020.

[197] X. Yao, X. Feng, J. Han, G. Cheng, and L. Guo, "Automatic weakly supervised object detection from high spatial resolution remote sensing images via dynamic curriculum learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 1, pp. 675–685, 2020.

[198] X. Feng, J. Han, X. Yao, and G. Cheng, "Tcanet: Triple context-aware network for weakly supervised object detection in remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 8, pp. 6946–6955, 2020.

[199] X. Feng, X. Yao, G. Cheng, J. Han, and J. Han, "Saenet: Self-supervised adversarial and equivariant network for weakly supervised object detection in remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–11, 2021.

[200] Z. Huang, Y. Zou, B. V. K. V. Kumar, and D. Huang, "Comprehensive attention self-distillation for weakly-supervised object detection," in *Advances in Neural Information Processing Systems* (H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, eds.), vol. 33, pp. 16797–16807, Curran Associates, Inc., 2020.

[201] V. Kantorov, M. Oquab, M. Cho, and I. Laptev, "Contextlocnet: Context-aware deep network models for weakly supervised localization," in *Computer Vision – ECCV 2016* (B. Leibe, J. Matas, N. Sebe, and M. Welling, eds.), (Cham), pp. 350–365, Springer International Publishing, 2016.

[202] Z. Ren, Z. Yu, X. Yang, M.-Y. Liu, Y. J. Lee, A. G. Schwing, and J. Kautz, "Instance-aware, context-focused, and memory-efficient weakly supervised object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10598–10607, 2020.

[203] Y. Shen, R. Ji, S. Zhang, W. Zuo, and Y. Wang, "Generative adversarial learning towards fast weakly supervised detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[204] Y. Shen, R. Ji, K. Yang, C. Deng, and C. Wang, "Category-aware spatial constraint for weakly supervised detection," *IEEE Transactions on Image Processing*, vol. 29, pp. 843–858, 2020.

[205] A. Arun, C. V. Jawahar, and M. P. Kumar, "Dissimilarity coefficient based weakly supervised object detection," 2018.

[206] F. Wan, C. Liu, W. Ke, X. Ji, J. Jiao, and Q. Ye, "C-mil: Continuation multiple instance learning for weakly supervised object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[207] X. Zhang, J. Feng, H. Xiong, and Q. Tian, "Zigzag learning for weakly supervised object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[208] F. Zhang, B. Du, L. Zhang, and M. Xu, "Weakly supervised learning based on coupled convolutional neural networks for aircraft detection," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 9, pp. 5553–5563, 2016.

[209] J. Ji, T. Zhang, Z. Yang, L. Jiang, W. Zhong, and H. Xiong, "Aircraft detection from remote sensing image based on a weakly supervised attention model," in *IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium*, pp. 322–325, IEEE, 2019.

[210] J. Han, D. Zhang, G. Cheng, L. Guo, and J. Ren, "Object detection in optical remote sensing images based on weakly supervised learning and high-level feature learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 6, pp. 3325–3337, 2014.

[211] G. Cheng, J. Han, P. Zhou, and L. Guo, "Scalable multi-class geospatial object detection in high-spatial-resolution remote sensing images," in *2014 IEEE Geoscience and Remote Sensing Symposium*, pp. 2479–2482, IEEE, 2014.

[212] P. Zhou, D. Zhang, G. Cheng, and J. Han, "Negative bootstrapping for weakly supervised target detection in remote sensing images," in *2015 IEEE International Conference on Multimedia Big Data*, pp. 318–323, IEEE, 2015.

[213] P. Zhou, G. Cheng, Z. Liu, S. Bu, and X. Hu, "Weakly supervised target detection in remote sensing images based on transferred deep features and negative bootstrapping," *Multidimensional Systems and Signal Processing*, vol. 27, no. 4, pp. 925–944, 2016.

[214] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[215] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, vol. 1, pp. 886–893, Ieee, 2005.

[216] G. I. Webb, *Bayes' Rule*, pp. 99–99. Boston, MA: Springer US, 2017.

[217] R. Salakhutdinov and G. Hinton, "Deep boltzmann machines," in *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics* (D. van Dyk and M. Welling, eds.), vol. 5 of *Proceedings of Machine Learning Research*, (Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA), pp. 448–455, PMLR, 16–18 Apr 2009.

[218] Y. Li, Y. Zhang, X. Huang, and A. L. Yuille, "Deep networks under scene-level supervision for multi-class geospatial object detection from remote sensing images," *ISPRS journal of photogrammetry and remote sensing*, vol. 146, pp. 182–196, 2018.

[219] B. Aygüneş, S. Aksoy, and R. G. Cinbiş, "Weakly supervised deep convolutional networks for fine-grained object recognition in multispectral images," in *IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium*, pp. 1478–1481, IEEE, 2019.

[220] B. Aygunes, R. G. Cinbis, and S. Aksoy, "Weakly supervised instance attention for multisource fine-grained object recognition with an application to tree species classification," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 176, pp. 262–274, 2021.

[221] S. Chen, D. Shao, X. Shu, C. Zhang, and J. Wang, "Fcc-net: A full-coverage collaborative network for weakly supervised remote sensing object detection," *Electronics*, vol. 9, no. 9, p. 1356, 2020.

[222] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," ICML '09, (New York, NY, USA), p. 41–48, Association for Computing Machinery, 2009.

[223] E. Sangineto, M. Nabi, D. Culibrk, and N. Sebe, "Self paced deep learning for weakly supervised object detection," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 3, pp. 712–725, 2018.

[224] D. Zhang, J. Han, L. Zhao, and D. Meng, "Leveraging prior-knowledge for weakly supervised object detection under a collaborative self-paced curriculum learning framework," *International Journal of Computer Vision*, vol. 127, no. 4, pp. 363–380, 2019.

[225] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2921–2929, 2016.

[226] G. Cheng, X. Xie, W. Chen, X. Feng, X. Yao, and J. Han, "Self-guided proposal gen-

eration for weakly supervised object detection," *IEEE Transactions on Geoscience and Remote Sensing*, 2022.

[227] F. Wan, P. Wei, J. Jiao, Z. Han, and Q. Ye, "Min-entropy latent model for weakly supervised object detection," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1297–1306, 2018.

[228] J. Ahn, S. Cho, and S. Kwak, "Weakly supervised learning of instance segmentation with inter-pixel relations," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2204–2213, 2019.

[229] L. Du, H. Dai, Y. Wang, W. Xie, and Z. Wang, "Target discrimination based on weakly supervised learning for high-resolution sar images in complex scenes," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 1, pp. 461–472, 2019.

[230] W. Li, C.-Y. Hsu, and M. Hu, "Tobler's first law in geoai: A spatially explicit deep learning model for terrain feature detection under weak supervision," *Annals of the American Association of Geographers*, vol. 111, no. 7, pp. 1887–1905, 2021.

[231] P. Berg, D. Santana Maia, M.-T. Pham, and S. Lefèvre, "Weakly supervised detection of marine animals in high resolution aerial images," *Remote Sensing*, vol. 14, no. 2, p. 339, 2022.

[232] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[233] B. Wang, Y. Zhao, and X. Li, "Multiple instance graph learning for weakly supervised remote sensing object detection," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–12, 2021.

[234] G. Jocher, A. Stoken, J. Borovec, L. Changyu, A. Hogan, L. Diaconu, F. Ingham, J. Poznanski, J. Fang, L. Yu, *et al.*, "ultralytics/yolov5: v3. 1-bug fixes and performance improvements," *Version v3*, vol. 1, 2020.

[235] M. A. Malbog, "Mask r-cnn for pedestrian crosswalk detection and instance segmentation," in *2019 IEEE 6th International Conference on Engineering Technologies and Applied Sciences (ICETAS)*, pp. 1–5, 2019.

[236] W. Ge, W. Huang, S. Guo, and M. Scott, "Label-penet: Sequential label propagation and enhancement networks for weakly supervised instance segmentation," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3344–3353, 2019.

[237] B. Kim, Y. Yoo, C. Rhee, and J. Kim, “Beyond semantic to instance segmentation: Weakly-supervised instance segmentation via semantic knowledge transfer and self-refinement,” 2021.

[238] J. Lee, J. Yi, C. Shin, and S. Yoon, “Bbam: Bounding box attribution map for weakly supervised semantic and instance segmentation,” 2021.

[239] W. Li, Y. Chen, W. Liu, and J. Zhu, “Deep level set for box-supervised instance segmentation in aerial images,” 2021.

[240] Z. Tian, C. Shen, X. Wang, and H. Chen, “Boxinst: High-performance instance segmentation with box annotations,” 2020.

[241] I. H. Laradji, D. Vazquez, and M. Schmidt, “Where are the masks: Instance segmentation with image-level supervision,” 2019.

[242] Y. Liu, Y.-H. Wu, P. Wen, Y. Shi, Y. Qiu, and M.-M. Cheng, “Leveraging instance-, image- and dataset-level information for weakly supervised instance segmentation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 3, pp. 1415–1428, 2022.

[243] X. Yan, L. Shen, J. Wang, X. Deng, and Z. Li, “Msg-sr-net: A weakly supervised network integrating multiscale generation and superpixel refinement for building extraction from high-resolution remotely sensed imageries,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 15, pp. 1012–1023, 2022.

[244] M. Abbasi, S. Mostafa, A. S. Vieira, N. Patorniti, and R. A. Stewart, “Mapping roofing with asbestos-containing material by using remote sensing imagery and machine learning-based image classification: A state-of-the-art review,” *Sustainability*, vol. 14, no. 13, 2022.

[245] S. Shahab, M. Anjum, and M. S. Umar, “Deep learning applications in solid waste management: A deep literature review,” *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 3, 2022.

[246] M. Radkevich, F. Mukhammadaliyeva, K. Shipilova, N. Umarova, and A. Gapirov, “Land pollution by illegal dumps in the tashkent region,” *IOP Conference Series: Earth and Environmental Science*, vol. 1068, p. 012036, jul 2022.

[247] B. K. Mavakala, P. Sivalingam, A. Laffite, C. K. Mulaji, G. Giuliani, P. T. Mpiana, and J. Poté, “Evaluation of heavy metal content and potential ecological risks in

soil samples from wild solid waste dumpsites in developing country under tropical conditions," *Environmental Challenges*, vol. 7, p. 100461, 2022.

[248] G. Rocco, T. Petitti, N. Martucci, M. C. Piccirillo, A. La Rocca, C. La Manna, G. De Luca, A. Morabito, A. Chirico, R. Franco, *et al.*, "Survival after surgical treatment of lung cancer arising in the population exposed to illegal dumping of toxic waste in the land of fires ('terra dei fuochi') of southern italy," *Anticancer research*, vol. 36, no. 5, pp. 2119–2124, 2016.

[249] M. Kubásek and J. Hřebíček, "Involving citizens into mapping of illegal landfills and other civic issues in the czech republic," 2014.

[250] R. Jordá-Borrell, F. Ruiz-Rodríguez, and Á. L. Lucendo-Monedero, "Factor analysis and geographic information system for determining probability areas of presence of illegal landfills," *Ecological Indicators*, vol. 37, pp. 151–160, 2014.

[251] L. C. Quesada-Ruiz, V. Rodriguez-Galiano, and R. Jordá-Borrell, "Characterization and mapping of illegal landfill potential occurrence in the canary islands," *Waste Management*, vol. 85, pp. 506–518, 2019.

[252] M. Krówczyńska, E. Raczko, N. Staniszewska, and E. Wilk, "Asbestos—cement roofing identification using remote sensing and convolutional neural networks (cnns)," *Remote Sensing*, vol. 12, no. 3, 2020.

[253] D.-M. Seo, H.-J. Woo, M.-S. Kim, W.-H. Hong, I.-H. Kim, and S.-C. Baek, "Identification of asbestos slates in buildings based on faster region-based convolutional neural network (faster r-cnn) and drone-based aerial imagery," *Drones*, vol. 6, no. 8, 2022.

[254] K. Glanville and H.-C. Chang, "Remote sensing analysis techniques and sensor requirements to support the mapping of illegal domestic waste disposal sites in queensland, australia," *Remote Sensing*, vol. 7, no. 10, pp. 13053–13069, 2015.

[255] D. Garofalo and F. Wobber, "Solid waste and remote sensing," *Photogrammetric engineering*, vol. 40, no. 1, pp. 45–59, 1974.

[256] T. L. Erb, W. R. Philipson, W. L. Teng, and T. Liang, "Analysis of landfills with historic airphotos," *Photogrammetric Engineering and Remote Sensing*, vol. 47, no. 9, pp. 1363–1369, 1981.

[257] S. Silvestri and M. Omri, "A method for the remote sensing identification of uncontrolled landfills: formulation and validation," *International Journal of Remote Sensing*, vol. 29, no. 4, pp. 975–989, 2008.

[258] J. Gill, K. Faisal, A. Shaker, and W. Y. Yan, “Detection of waste dumping locations in landfill using multi-temporal landsat thermal images,” *Waste Management & Research*, vol. 37, no. 4, pp. 386–393, 2019.

[259] L. Selani, *Mapping Illegal Dumping Using a High Resolution Remote Sensing Image Case Study: Soweto Township in South Africa.* PhD thesis, University of the Witwatersrand, Faculty of Science, School of Geography . . . , 2017.

[260] C. Angelino, M. Focareta, S. Parrilli, L. Cicala, G. Piacquadio, G. Meoli, and M. De Mizio, “A case study on the detection of illegal dumps with gis and remote sensing images,” in *Earth Resources and Environmental Remote Sensing/GIS Applications IX*, vol. 10790, pp. 165–171, SPIE, 2018.

[261] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 2, pp. 318–327, 2020.

[262] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2261–2269, 2017.

[263] M. R. Devesa and A. V. Brust, “Mapping illegal waste dumping sites with neural-network classification of satellite imagery,” *arXiv preprint arXiv:2110.08599*, 2021.

[264] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” 2015.

[265] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, “On calibration of modern neural networks,” in *International conference on machine learning*, pp. 1321–1330, PMLR, 2017.

[266] R. N. Torres, P. Fraternali, and A. Biscontini, “On the use of class activation maps in remote sensing: the case of illegal landfills,” in *2021 IEEE 8th International Conference on Data Science and Advanced Analytics (DSAA)*, pp. 1–10, IEEE, 2021.

[267] J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7132–7141, 2018.

[268] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, “Eca-net: Efficient channel attention for deep convolutional neural networks,” pp. 11531–11539, 06 2020.

[269] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, “Cbam: Convolutional block attention

module," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 3–19, 2018.

[270] N. Karimi, K. T. W. Ng, and A. Richter, "Development and application of an analytical framework for mapping probable illegal dumping sites using nighttime light imagery and various remote sensing indices," *Waste Management*, vol. 143, pp. 195–205, 2022.

[271] M. Đidelija, N. Kulo, A. Mulahusić, N. Tuno, and J. Topoljak, "Segmentation scale parameter influence on the accuracy of detecting illegal landfills on satellite imagery. a case study for novo sarajevo," *Ecological Informatics*, vol. 70, p. 101755, 2022.

[272] A. Rajkumar, C. A. Kft, T. Sziranyi, and A. Majdik, "Detecting landfills using multi-spectral satellite images and deep learning methods,"

[273] C. Padubidri, A. Kamilaris, and S. Karatsiolis, "Accurate detection of illegal dumping sites using high resolution aerial photography and deep learning," in *2022 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops)*, pp. 451–456, IEEE, 2022.

[274] M. Gómez-Delgado and S. Tarantola, "Global sensitivity analysis, gis and multi-criteria evaluation for a sustainable planning of a hazardous waste disposal site in spain," *International Journal of Geographical Information Science*, vol. 20, no. 4, pp. 449–466, 2006.

[275] H.-Y. Lin and J.-J. Kao, "A vector-based spatial model for landfill siting," *Journal of Hazardous Materials*, vol. 58, no. 1-3, pp. 3–14, 1998.

[276] A. Alfarrarjeh, S. H. Kim, S. Agrawal, M. Ashok, S. Y. Kim, and C. Shahabi, "Image classification to determine the level of street cleanliness: A case study," in *2018 IEEE Fourth International Conference on Multimedia Big Data (BigMM)*, pp. 1–5, IEEE, 2018.

[277] A. Dabholkar, B. Muthiyan, S. Srinivasan, S. Ravi, H. Jeon, and J. Gao, "Smart illegal dumping detection," in *2017 IEEE Third International Conference on Big Data Computing Service and Applications (BigDataService)*, pp. 255–260, IEEE, 2017.

[278] G. Mittal, K. B. Yagnik, M. Garg, and N. C. Krishnan, "Spotgarbage: smartphone app to detect garbage using deep learning," in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pp. 940–945, 2016.

[279] Y. Wang and X. Zhang, "Autonomous garbage detection for intelligent urban management," in *MATEC Web of Conferences*, vol. 232, p. 01056, EDP Sciences, 2018.

[280] B. De Carolis, F. Ladogana, and N. Macchiarulo, "Yolo trashnet: Garbage detection in video streams," in *2020 IEEE Conference on Evolving and Adaptive Intelligent Systems (EAIS)*, pp. 1–7, IEEE, 2020.

[281] K. Yun, Y. Kwon, S. Oh, J. Moon, and J. Park, "Vision-based garbage dumping action detection for real-world surveillance platform," *ETRI Journal*, vol. 41, no. 4, pp. 494–505, 2019.

[282] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," 2018.

[283] S. Hochreiter, Y. Bengio, P. Frasconi, J. Schmidhuber, *et al.*, "Gradient flow in recurrent nets: the difficulty of learning long-term dependencies," 2001.

[284] X. Wang, S. Wang, C. Ning, and H. Zhou, "Enhanced feature pyramid network with deep semantic embedding for remote sensing scene classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 9, pp. 7918–7932, 2021.

[285] M. Rahimzadeh, A. Attar, and S. M. Sakhaei, "A fully automated deep learning-based network for detecting covid-19 from a new and large lung ct scan dataset," *Biomedical Signal Processing and Control*, vol. 68, p. 102588, 2021.

[286] J. M. Johnson and T. M. Khoshgoftaar, "Survey on deep learning with class imbalance," *Journal of Big Data*, vol. 6, no. 1, pp. 1–54, 2019.

[287] G. B. Dantzig, "Maximization of a linear function of variables subject to linear inequalities," *Activity analysis of production and allocation*, vol. 13, pp. 339–347, 1951.

[288] J. Lafferty, A. Mccallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," *Proceedings of the Eighteenth International Conference on Machine Learning*, pp. 282–289, 01 2001.

List of Figures

List of Tables

List of Abbreviations

Abbreviation	Long word/phrase
AE	Autoencoder
AI	Artificial Intelligence
AP	Average Precision
BB	Bounding Box
BCE	Binary Cross-Entropy
BiGAN	Bidirectional Generative Adversarial Network
CAM	Class Activation Map
CNN	Convolutional Neural Network
CRF	Conditional Random Field
CV	Computer Vision
DL	Deep Learning
ECE	Expected Calibration Error
FC	Fully Connected
FN	False Negative
FP	False Positive
FPN	Feature Pyramid Network
FS	Full Supervision / Fully Supervised
FSOD	Fully Supervised Object Detection
GAN	Generative Adversarial Network
GAP	Global Average Pooling
GIS	Geographic Information System
GSD	Ground Sample Distance
GT	Ground Truth
HS	Hyperspectral
HSI	Hyperspectral Image

IoU	Intersection Over Union
IS	Instance Segmentation
MAE	Masked Autoencoder
mAP	Mean Average Precision
MCE	Multi Criteria Evaluation
MCG	Multi-scale Combinatorial Grouping
MIL	Multiple Instance Learning
ML	Machine Learning
NIR	Near Infrared
OD	Object Detection
PR Curve	Precision-Recall Curve
RGB	Red, Green, Blue
RPN	Region Proposal Network
RS	Remote Sensing
RSI	Remote Sensing Image
RSWSOD	Remote Sensing Weakly Supervised Object Detection
Sb-SaS	Saliency-based Self-adaptive Segmentation
SW	Sliding Window
SAR	Synthetic Aperture Radar
SDA	Synthetic Data Augmentation
SDG	Sustainable Development Goals
SOTA	State-Of-The-Art
SS	Semantic Segmentation
SSL	Self-Supervised Learning
SVM	Support Vector Machine
TDL	Target Detector Learning
TL	Transfer Learning
TN	True Negative
TP	True Positive
TSI	Training Set Initialization
UAV	Unmanned Aerial Vehicle
VAE	Variational Autoencoder
WS	Weak Supervision / Weakly Supervised

WSIS	Weakly Supervised Instance Segmentation
WSL	Weakly Supervised Learning
WSOD	Weakly Supervised Object Detection
WSOL	Weakly Supervised Object Localization
WSSS	Weakly Supervised Semantic Segmentation