POLITECNICO DI MILANO
School of Industrial and Information Engineering
Department of Electronics, Information and Bioengineering
Master of Science in Computer Science Engineering

A system for annotating and analysing multi-source geo-referenced images for environmental applications

Supervisor: Piero Fraternali
Co-supervisor: Rocio Nahime Torres

Master Thesis of:
Alberto Federico Pagani, 837576

Academic Year 2018-2019

Dedicated to my family.

Acknowledgements

At the end of a journey it is important to take a moment to reflect on where it started and more importantly what happened during it.

For this very reason I want to first thank my family for their endless support throughout my life, they always believed in me in every moment, specially in the hardest ones in which they did it even more than me. During the thesis they tirelessly encouraged me and really helped me to going on and reach my goal. I will never be able to thank them enough for this.

I want to say special thanks to my friend and colleague Alessandro for his friendship during this incredibly journey that we did together, without him it would not have been the same.
I also want to thank all the friends and colleagues who have been part of my life in these years.

I want to thank Prof. Piero Fraternali for this opportunity that allowed me to work with him and his team.

Finally I want to thank Rocio Nahime Torres that assisted me during this thesis and taught me so much.

Sommario

Il problema dello smaltimento illegale dei rifiuti oggigiorno rappresenta una grave minaccia all'ambiente in tutto il mondo e continua ad aggravarsi. Per essere fatto correttamente lo smaltimento dei rifiuti deve seguire specifiche regole pensate per proteggere la salute delle persone. Tuttavia si verifica sempre più spesso che privati cittadini gettino illegalmente piccole quantità di rifiuti e che grandi organizzazioni criminali creino vere e proprie discariche abusive di grosse dimensioni. Per trovare una soluzione è necessario sfruttare un'analisi mirata, e possibilmente automatica, che porti all'individuazione di questi siti critici. Un primo passo in questa direzione è rappresentato dall'analisi dati e dalla creazione di un dataset che possa fare da base per l'implementazione di metodi automatici di identificazione e per la loro validazione. Per realizzare un dataset di entità geo-referenziate con le loro rispettive descrizioni non solo è necessaria la disponibilità di dati ma anche di strumenti che possano semplificare il processo e renderlo efficiente. In questo progetto presentiamo uno strumento utilizzato per l'annotazione di immagini geo-referenziate da sorgenti diverse, quali satellitari e ortofotografiche. Lo strumento è stato sviluppato in modo da poter essere adattato e reso compatibile anche con casi d'uso generici e scopi diversi da quello proposto in questo lavoro di tesi. Oltre a questo sono stati realizzati anche altri strumenti utili al reperimento delle immagini satellitari e all'elaborazione di immagini ortofotografiche usate nello strumento di annotazione. Gli strumenti sviluppati sono stati usati per l'analisi di più di 1500 immagini che hanno portato alla realizzazione di un dataset di più di 10900 annotazioni, ciascuna delle quali consiste in un poligono corredato di una etichetta che lo descriva. Queste annotazioni sono state realizzate identificando discariche abusive sospette e i loro elementi più caratterizzanti.

Abstract

The problem of illegal waste disposal nowadays it is a serious and worldwide threat to the environment that continues to grow. To be done correctly, waste disposal has to follow specific rules designed to protect people's health. However it is increasingly common for private citizens to illegally dispose of small quantities of waste and for large criminal organizations to create real large illegal landfills. To find a solution, it is necessary to take advantage of an intelligent analysis, possibly automatic, that leads to the identification of these critical sites. A first step in this direction is represented by data analysis and by the creation of a dataset that could represent the basis for the implementation of automatic identification methods and for their validation. To create a dataset of geo-referenced entities with their respective descriptions, not only is necessary the availability of data but also of tools that can simplify the process and make it efficient. In this project we present a tool for annotating geo-referenced images from different sources, such as satellites and orthophotos. The tool has been developed so that it can be adapted and made compatible even with generic use cases and purposes other than those proposed in this thesis work. In addition to this, other useful tools have been created for retrieving satellite images and processing orthophoto images used in the annotation tool. The developed tools have been used for the analysis of more than 1500 images which led to the creation of a dataset of more than 10900 annotations, each of which consists of a polygon with its corresponding and describing label. These annotations have been realized identifying suspicious illegal landfills and their characterizing elements.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The disposal of wastes to be safe for the environment must be done in specific structures that are equipped to correctly and safely dispose the wastes but not always this is done. This causes the formation of landfills that actually do not have sewage disposal systems and all the required safety systems that are regulated for an authorized landfill and thus being illegal. The presence of this illegal landfills represents a huge treat for the environment. Nowadays the issue of illegal landfills is a trend on the rise that is becoming really problematic, all over the world, causing a lot of damage not only to the environment, both vegetation and animals, but also to people's health[1]. There are many reasons that cause this awful issue, probably the main one is avoiding to pay disposal fees and other fees related to the disposal of special wastes. The possible approaches usable to stop it are:

- **prevention**: aimed to find areas that are most suitable to become an illegal landfill
- **detection and monitoring**: aimed to find areas that actually are illegal landfills and monitor their evolution over time
- **field work**: aimed to actually verify the existence of an illegal landfill in a certain area

Field work as well as other usually applied techniques for landfill identification typically require a large amount of manual human work, e.g. manual visual inspection by expert personnel of images from remote sensing technologies. For this reason field works techniques are very costly solutions to the illegal dumping issue and cannot represent a potential definitive solution to it.

Remote sensing (RS) technologies, along with Geographical Information System (GIS), allowed the use of techniques able to reduce the need of field work since they enable a more structured work on the RS data, e.g. satellite images and radar observations, making them best suited for detection[2][3] and monitoring[3] tasks also on large areas. However there are cases in which their results need to be physically checked requiring field work[2][4] or can yield false positives depending on other environmental reasons, like in cases where the methodology relies on finding vegetation stress that could be caused by something different from an illegal landfill like[3].

In the objective of creating more structured and less time consuming methodologies, methods from the machine learning field, based on large of amount of GIS data and data from the local authorities archives[5], and the deep learning field, capable to automatically detect objects inside images[6][7] and already tested in the field of wastes detection[8], have proven to be successful approaches which led us to believe their potential in automating the illegal dumping detection.

What this research wants to propose, as shown in Figure 1.1, is the starting point for a new method for autonomous illegal landfills detection that, exploiting RS imagery like aerial orthophoto or satellite images in the optical spectrum, relies on the creation of a multi class dataset of annotated images containing everything needed by a deep learning method for computer vision (CV) to learn how to detect illegal landfills.
This new proposal works with these major steps:

1. Imagery retrieval and preprocessing, to provide georeferenced imagery

2. Imagery classification and annotation, through a custom crowdsourcing web annotation tool

3. Usage of the produced annotations to create a dataset that can be later

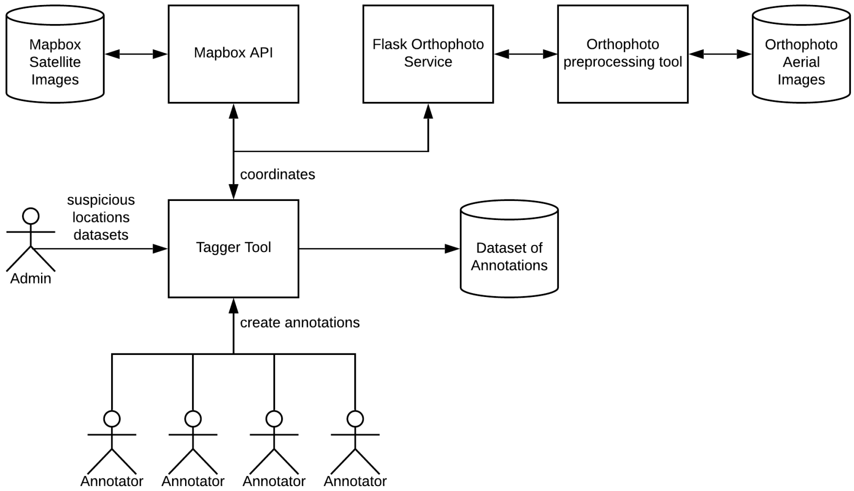

Figure 1.1: Project pipeline.

used for deep learning methods for CV

In our proposed solution we used satellite images from Mapbox map provider and aerial orthophoto images, provided us by Agenzia Regionale per la Protezione Ambientale (ARPA) Lombardia. For the satellite images it was needed to develop a custom crawler able to download georeferenced images from Mapbox in order to create the satellite images pool. For the aerial orthophoto images instead it has been necessary to develop a specific preprocessing tool able to produce small georeferenced images from larger images also in a different Coordinate Reference System (CRS) and then a web service that returns the preprocessed images allowing the web annotation tool to retrieve them making a request with coordinates and distance, that represents a measure of the area that the image has to cover.

To effectively create a dataset large enough to be successfully used for deep learning detection tasks we decided to develop a crowdsourcing web annotation tool, specifically for this case scenario, that allows many users to cooperatively work in classifying and annotating georeferenced images. This has been necessary because crowdsourcing-like approaches are probably the only ones that can reliably offer large amount of working hours in a small

time window as demonstrated also by other huge datasets built in the deep learning community like ImageNet[9] and Microsoft COCO[10] that exploited for the most Amazon Mechanical Turk[1] workers.
This crowdsourcing web annotation tool allows to:

- **create campaigns** from list of already suspicious sites
- let workers **cooperatively** evaluate suspicious site locations and **annotate wastes elements** found inside them
- let admin **inspect campaigns sites on maps** allowing a greater understanding of their territorial distribution
- **elaborate statistics** for the created campaigns
- **extract the results** produced by worker in both geographic and deep learning ready formats

Following the objectives of this study, this thesis has been divided in the following chapters. Chapter 2 will discuss the state of the art of the methodologies used for prevention, detection and monitoring of illegal landfills that brought to the development of this project. Chapter 3 will cover everything about the data sources from which the images are retrieved and the properties of the dataset. Chapter 4 will explain in every detail the specifications and the capabilities of the tools developed, with particular focus for the web annotation tool used to create the dataset. Chapter 5 will show a detailed workflow using the web annotation tool and the dataset results obtained from the annotation process. Chapter 6 will summarize the overall study with its achievements and explore the possible evolution and future works that could arise from it.

[1]https://www.mturk.com

Chapter 2

Related works

In this chapter we provide a review to the state of the art for illegal landfills identification, deep learning methods for computer vision applied to similar use cases to motivate the viability of the proposed approach and also a short report about dataset notion, with examples of dataset for remote sensing imagery, and crowdsourcing tasks.

2.1 Methods for illegal landfills identification

The analysis of illegal landfill is not only to identify a new site or predict the probability of finding a site in certain area but also can be used for monitoring already existing ones and this methodologies can divided in different categories depending on the underlying technologies and data used.

Prior to the advent of technologies and methodologies that enabled different and more efficient workflows field work represented for long time the first type of approach for illegal landfill identification, however this not without great drawbacks like its costs and very time consuming activities. For this reason it is crucial to find new methodologies that allows to solve this important drawbacks. An example of methodologies that exploit very much field work is represented by geophysical techniques that are often used to help the field work when the wastes are buried like in the case of Ground Penetrating Radar (GPR)[11] and also provide ground truth data, due the reliability of their results, like in [2]. Field work and geophysical methodologies shares also another drawback represented by the small scale on which they are applicable, trying to apply them on very large areas results in very high costs and also require an amount of time not feasible.

Thanks to technological advancements in sensors field, Remote Sensing (RS) technologies improved their capabilities to produce high quality data becoming relevant as a potential enhancement to field work techniques. Remote sensing techniques base their work on data provided by active or passive remote sensors[12] that do not require to be on site to collect them, since these sensors are commonly used satellite and airborne. For this reason RS technologies have become the activators of many methodologies, like for example[13][3] and [14], that also exploit Geographical Information System (GIS), that have the great advantage to partially overcome the necessity of field work, if not in cases like [2] where field work is required to gather a valid ground truth or cases like [14] where the approach is meant to offer priority lists of the areas that are most suited to contains contaminated sites and on which field work is profitable. Another great advantage is the possibility easily work on a large scale area like in[14][2], feature actually not available for field work techniques that have to manually parse all the area with the work of highly specialized personnel.

As long as illegal dumping issue kept worsening and both machine learning techniques and crowdsourcing approaches gained importance new methodologies, like[5][15], were tested to exploit the large amount of RS and GIS data available with the aim to enable simplified and autonomous larger scale works. When also deep learning methods for Computer Vision (CV) gained attention enabling the detection with good accuracy of objects inside images[16][6][7] new methodologies like[8], able to exploit their capabilities, have been developed and experimented. Methodologies using deep learning CV algorithms allowed the autonomous detection of wastes inside non satellite images, thus still not very scalable, requiring less and less human time work for the detection task[8][17]. Also the creation of dataset from RS satellite imagery have been experimented[18] showing the possibility to use RS imagery to effectively create datasets for CV detection tasks.

The possibility to have RS imagery available along with promising results showed by CV methods and crowdsourcing approaches made us think to experiment our methodology oriented to the creation of a dataset for waste detection from RS imagery using a crowdsourcing approach.

2.1.1 Detection using remote sensing methodologies

Remote sensing (RS) refers to a set of technologies and techniques for survey and data collection that allows the retrieval of large amount of high quality data, especially in areas where other types of data collection are not possible, like field work for example.

Remote sensing technologies are divided in two major families, as shown in Table 2.1

Many active remote sensing technologies are based on Radar like observations and have great capabilities in the exploration of areas that will be otherwise very difficult to cover. These technologies offer the possibility to potentially cover almost every area of the Earth. A few examples of active sensors technologies is represented by Synthetic Aperture Radar (SAR) and Light Detection and Ranging (Lidar) as can be seen in Table 2.2.

Many passive remote sensing technologies are based on satellite imaging on different spectra, this aspect is another important and key feature offered by remote sensing technologies. The ability to exploit many different spectra allows to obtain data that offers different point of view of what is observed, in Table 2.3 it can be seen a few examples of technologies that possess different spectra capabilities and their possible example sources.

Sensors type	Description
Active sensors	Active remote sensors provide their own energy source emitting a beam of energy directed toward a target of interest and then measuring the reflected or back-scattered signal. All the measurements done depend on the time the signal takes to return as well as its returned amplitude and wavelength. With this technique it is possible to understand things like distance(location and shape can be extrapolated) and speed of an object. Some most commonly used active remote sensing techniques are Synthetic-Aperture Radar (SAR) and Light Detection and Ranging (LiDAR).[12]
Passive sensors	Passive remote sensors detect energy that occurs naturally in the environment, a vast majority of the energy is in the form of electromagnetic waves reflected off of the Earth from the Sun. Resulting images are measurements of sunlight reflected by the terrain. [12]

Table 2.1: Remote sensing families depending on sensor type

Active sensors	Description
SAR	Uses active sensors that work with a state of relative motion of its antenna with respect to the target region in order to provide distinctive long-term coherent-signal variations that are used to create finer spatial resolution that for example are useful to obtain good 3D surfaces of the target location.[12]
LiDAR	Uses active sensors that make optical measurements of scattered light to find the target. These measurements are made with airborne-sensors or even land-based sensors. The distance, and thus the position, of an object depends on the time delay between transmission and detection of a laser pulse. The accuracy is around 0.1m. This technology can be used to measure the land surface elevation beneath some medium, for example the vegetation.[12]

Table 2.2: Active sensors types

Sensors Spectra	Description	Examples
Visible light	The sensor used to create images based on visible light spectrum is a sensor able to capture three specific spectral wavelength ranges represented by: blue (452-512nm), green (533-590nm), red (636-673nm). The resulting imaging are high quality colored images, with high resolution, that lets see the target area in very similar way like the eye will do. These kind of images can be used to create high quality maps that can be used for applications like web maps, navigation systems and tasks for the elaboration of very high quality georeferenced images of specific areas.[19]	LANDSAT 8 Bands 2-3-4
Continues on the next page 2.3		

Continuation of Table 2.3		
Sensors Spectra	Description	Examples
Panchromatic	The sensor is a single channel detector that is sensitive to radiation within a broad wavelength range. The physical quantity measured is the apparent brightness of the targets. The spectral information, hence the color, of the targets is captured in this case. These images have a very high signal-noise ratio and enable the acquisition of very high quality mapping, actually sharper than the one available with visible light. Panchromatic images are also used in combination with low-res multi-spectral images to actually produce pan-sharpened images that are high-res colored images.[19]	IKONOS-PAN HRV-PAN
Continues on the next page 2.3		

Continuation of Table 2.3		
Sensors Spectra	Description	Examples
Multi-spectral	The sensor is a multi-channel detector with few spectral bands. Each channel is sensitive to radiations within a narrow wavelength band. The resulting image is multilayer and contains both the brightness and spectral information of the targets being observed. Sensor operating in the multi-spectral bands typically detects radiations in these wavelength bands: blue (450-500nm), green (500-590nm), red (610-680nm) and near infrared (790-890nm, NIR) bands[19]. Having access to NIR makes possible to study vegetation stress inside of an area.	LANDSAT-MSS IKONOS-MS
Super-spectral	A super-spectral sensor has many more spectral channels than a multi-spectral sensor. The bands are narrower and thanks to this it enables to capture the finer spectral characteristics of the targets, greatly improving the analysis possibilities given by multi-spectral sensors.[19]	MODIS
Continues on the next page 2.3		

Continuation of Table 2.3		
Sensors Spectra	Description	Examples
Hyper-spectral	A hyper-spectral imaging system is also known as an "imaging spectrometer". It acquires images in about a hundred or more contiguous spectral bands. The precise spectral information contained in a hyper-spectral image enables better characterisation and identification of targets. Hyper-spectral images having such narrow spectral bands make possible to extract precise features, available only at specific wavelengths, containing very specific information that can be used for example in vegetation stress or oceanic floor studies or even to study particular gas emissions from terrain.[19]	EO1-Hyperion sensor

Table 2.3: Remote sensing examples with related usage.

Remote sensing technologies have been widely used in these years to help in solving the rising issue of illegal landfills since they allow to exploit high resolution and multi-spectral images and also spatial resolution data to perform analysis of the terrain capable of detecting potential illegal sites[3][4].

All the spectral sensor technologies, just reviewed, have shown great potential for creating mapping of areas[20], remotely from satellite or other airborne approaches, and among these mapping methodologies available it is also possible to exploit thermal bands that actually gives a great insight of the temperature of an area. A research study [13] decided to exploit multi-temporal LANDSAT multi-spectral, atmospherically corrected using ATCOR2 model, and thermal bands to compute a Land Surface Temperature (LST) map of certain areas to investigate the relationship that exists between the gas emissions of a landfill and surface temperature of an area in its vicinity over the time. This case study started studying two known municipal non-hazardous landfills:

- the Trail Road landfill site located in Ottawa city, Ontario, Canada ($2km^2$ area)
- the Al-Jleeb landfill site located in the city of Al-Farwanyah, Kuwait ($5.5km^2$ area)

Their proposed methodology uses multi-spectral images to derive from thermal bands the LST used to perform a comparison with the air temperature of the landfill area. In the Trail Road landfill the comparison has been done with temporal data from the 2007 to 2008 years range, for that time range also the measurements of methane (CH4) from monitoring wells were available and thus used. Results for this comparison showed that there is a thermal difference between LST and the surrounding air temperature and also that exists a direct relationship between the increased methane emission over the years and the temperature difference, thus confirming their initial hypothesis. Also in the Al-Jleeb landfill comparison results have shown that exists a temperature difference between the LST and the air temperature in vicinity of landfill also pointing out that the difference decreases during the winter season, finding as explanation that since the methane emission is caused mainly by decomposition processes it possible that these kind of processes slow down with cold weather. The overall results of this study has proven the initial hypothesis of a direct relationship that exists between the gas emissions and LST of a landfill.

These results were later tested in research study [2], from the same authors, that used LST to detect gas emissions to locate the Jeleeb Al-Shuyoukh landfill, inside Kuwait. The specific location of the landfill is unknown because its exact dumping locations were lost during the Gulf war. The area considered in the process covered approximately $5.5km^2$ with data from a temporal span of ten years from 1985 to 1994. Many LST maps have been created from different time indexes in the span and then confronted together to detect the possible dumping locations. As ground truth, to assess the quality of the results they acquired from local authorities 50 boreholes gas emission measurements effectively able to determine the local concentration of gas emitted by the terrain (up to near 30m underground) verifying the potential presence of underground wastes. The comparison of the results with the ground truth data have shown a 72% recognition accuracy thus proving the possibility not only to study known landfill but also to use this approach for detection purposes on large areas.

In a case study [3] researchers analysed the hypothesis that the presence of a landfill can impact the vegetation stress of the green areas in its vicinity proposing the possibility to use it as a possible indicator of an illegal landfill presence; the study was carried out in Italy, specifically inside the Campania region in the provinces of Napoli and Caserta. This decision comes from the fact that specific spectral bands like Near InfraRed (NIR) and RED Very Near InfraRed (RED-VNIR), acquirable from multi and hyper-spectral sensors, are actually able to produce images that can show the health status of the plants and trees in a certain area making possible to understand if they are suffering or even dying for some reasons. Their project proposes the use of SIMDEO (Sistema Integrato per il Monitoraggio delle Discariche con dati EO) that aims to detect new illegal landfills and then monitor them. For the detection of possible leachates, caused by illegal landfills present in the area, they started from RED-VNIR data coming from RAPIDEYE and SPOT-5 images and built a new index called DDI (Dump Detection Index) capable to combine vegetation stress with images textural features. The DDI was tested on a list of well-known landfills and results proved the possible existence of vegetation stress caused by the closeness to the landfill. To verify even more the results they used ASTER-TIR (Thermal Infrared Spectroscopy) night-time images to compute the LST (Land Surface Temperature) in order to compare the results from both approaches; this verification confirmed the results obtained from the DDI showing that also in night time images it was possible to spot the landfill, mainly because uncontrolled landfills do not posses a proper methane collecting system and thus the gas dispersion cause a general temperature increase in the area. For the monitoring phase also SAR, in conjunction to optical data, were used to create both a 2D and 3D mapping able to effectively monitors variation in already known landfills, that is also more consistent than an only-optical data approach since is more resistant to saturation and bad weather. This monitoring approach tested with temporal data on a known closed landfill was able to spot variations in height and volume of the landfill actually proving that the landfill was still active in the range of years taken into account. Overall this two approaches from SIMDEO project have been able to point out the great potential of using remote sensing technologies in illegal landfills detection and monitoring, also showing that vegetation stress is a possible good indicator of potential illegal landfills in areas where they should not be present.

A research study [4] from Japan studied the possibility to use remote sensing technologies for monitoring landfills or even more generically wastes deployed on land areas doing human visual inspection of satellite images from different sensor sources.

The sensor sources took in consideration are the on-board equipment of ALOS (Advanced Land Observing Satellite) and Quickbird satellite. As study area they chose three known landfills for the ALOS imagery and one reconstructed university campus plus one known landfill for the Quickbird imagery. These choices also provided ground truth data used to compare the visual inspection done with the imaging since there are available public data from authorities and ground observations of these specific areas.

For the ALOS datasets they demonstrated that is effectively possible to use imagery to visually detect wastes, also distinguishing them for the surrounding vegetation, in particular using pan-sharpened images from PRISM (Panchromatic Remote-sensing Instrument for Stereo Mapping) and AVNIR-2 (Advanced Visible and Near Infrared Radiometer type 2) sensors. Instead from their observations done with PALSAR (Phased Array type L-band Synthetic Aperture Radar) they pointed out that its intensity imagery is not suitable for detection of waste disposal sites.

For the Quickbird datasets they demonstrated that these pan-sharpened images are really effective in allowing visual detection also of small wastes, even surrounded by vegetation, larger than 2x2m in particular pointing out that:

- multi-spectral data greatly improves the capability to discriminate between waste and vegetation because of their different spectral characteristics
- iron scraps and plastic waste under the 2x2m are difficult to detect on bare soil because it exhibits higher reflectance

What has been pointed out in this study is very important because lays the foundation of possible works that use optical images for visual recognition of wastes and illegal landfills.

An interesting study [1] from Australia made a very specific review of the available methods for mapping illegal waste disposal sites discussing their applicability on illegal domestic waste disposal.
They reviewed with respect to the sensors first and then to the methodologies and their potential applicability for monitoring of small sized domestic wastes, that have volumes of 200 liters or more.

Regarding the applicability of sensors they concluded that the use of optical or panchromatic imagery in low resolution, multi-spectral and SAR observation has not proven its usefulness in monitoring domestic wastes, probably because more oriented to be used for methodologies tuned up to larger scale detection. Instead sensors that produce very high resolution panchromatic imagery are very likely to be exploited by methodologies to monitor small sized waste like the domestic ones.

Regarding the applicability they pointed out the issues regarding human visual based identification and the success obtained from maximum likelihood classification methodology. Visual based identification methods using very high resolution imagery can be effective in detecting small sized wastes even though they come to a great expense of time and expertise since they require constant human work to let them work and because of this not really recommended in a large scale scenario. Maximum likelihood classification methodology, like[21], when used in conjunction to high resolution imagery are effectively. Moreover they also noted that additional use of GIS analysis with remote sensing methodologies can effectively improve the accuracy of the map produced thus confirming that GIS mixed approaches are useful.

Different remote sensing methodologies for illegal landfills identification and monitoring proved to be successfully, these methodologies along with other successful approaches that used optical images for visual inspection motivated us in thinking that this two methodologies together in association with computer vision algorithm worth to be tested for the illegal dumping issue.

2.1.2 Detection using statistical predictive methodologies

Thanks to the development of machine learning field the predictive methods, that relies in the identification of the underlying statistical model, have become more and more present in many fields where they have been experimented to test if these kind of approaches can be meaningful compared to the already known in-field approaches.

A case study coming from Spain [5] tested a logistic regression methodology to effectively produce a predictive model usable to accurately predict the presence of an illegal landfill based on the geographical, social-economical, legal and cultural variables that characterize the territory taken in analysis. The study comprehended three major phases:

1. Identification of the existing relationships between spatial and behavioural parameters (independent variables) that influences the presence of illegal landfills
2. Measurement of the magnitude of the relationships found
3. Establishment, by means of a logistic regression equation, of the probability that an uncontrolled landfill will appear according to the spatial and behavioural variables included in the final model

For each group of independent variables taken into account the researchers proposed some hypotheses:

1. The occurrence of illegal landfills is conditioned by the topographical characteristics and land use of the site
2. The occurrence of illegal landfills is conditioned by the accumulation of construction and demolition waste in easily accessible and unsupervised areas.
3. Uncontrolled landfills are more likely to appear near urban areas and close to secondary communication routes

4. The occurrence of illegal landfills is associated with municipalities with larger populations and higher per capita income and a greater number of companies in the industrial and construction sectors.

5. The implementation of waste management systems (facilities) and municipal policies reduces the occurrence of illegal landfills

6. The population's lack of environmental awareness has a significant influence on the occurrence of illegal landfills

All the data that they used to compute the theoretical model that brought to the listed above hypotheses have been obtained through: field work, elaboration of spatial and geographical variables on GIS and from official statistics of municipalities. Every hypothesis has been tested with partial logistic regressions analysis and one last joint analysis brought to the final model. The final model confirmed all the derived hypotheses and when assessed it proved an high accuracy both in prediction of non illegal landfills areas with 92,9% and illegal landfills areas with 93,5%, thus obtaining an overall accuracy of 93.1% that represents a remarkable result. Moreover this study confirmed the possibility to use machine learning tools to improve the predicting capabilities of currently known methodologies.

Another study that exploited statistical methodologies applying it to RS imagery to predict the presence of illegal landfills is represented by [21] from Italy. In this study the authors started from the hypothesis that stressed vegetation could be a good indicator for the presence of potential illegal landfills. Their proposed approach used IKONOS multi-spectral pansharpened images, previously atmospherically corrected, to select 13 regions of interest covered by vegetation stress whose spectra have been used to calibrate a straightforward Maximum Likelihood method that allowed them to classify the stressed vegetation cover of the study area, 1969km^2 in the NE of Italy in the Venice lagoon. The spectra were then merged in 4 classes:

- Class 1: very stressed brown–yellow vegetation cover
- Class 2: non-uniformly stressed vegetation cover

- Class 3: bare soil with a very low vegetation presence
- Class 4: lightly stressed vegetation cover

From the data resulted in the classification of the area followed a human/visual interpretation and digitization process, for the site that resulted only partly classified, and finally a step of selection to exclude the low probability ones following two criteria:

- Areas not accessible through streets or paths cannot be a disposal site
- Sites for which no previous suspect activity could be traced from historical aerial photographs (when available) or from reports of the local authorities were excluded.

Their results proved the effective feasibility of using an hybrid approach that unifies statistical prediction methods (using RS data), manual/human work and also the relevance of using vegetation stress as possible illegal landfill indicator. Moreover their result pointed out the necessity of very high resolution and quality images to obtain good result as later confirmed by another study[1] that reviewed their methodologies for small sized waste detection.

A case study[14], from Italy, starting from the good results of study[21] in integrating statistical methods with RS data proposed an innovative approach that used a combination of Multi-Criteria and Multi-Factor Evaluation (MCE and MFE) with GIS. Also this study studied the area in the NE of Italy inside the Venice lagoon. Their proposed approach was to identify a selection of:

- siting factors, responsible for the suitability of certain area to contain an illegal landfill
- siting criteria, responsible for the impossibility for landfills to be found in specific areas

With data resulting from combined MCE and MFE they produced, through GIS tools, suitability maps able to divide the study area in three probability ranges(red=high, green=medium, blue=low), depending on the probability to find an illegal landfill there. Validating the results from suitability maps with 19 know sites, from the validation set, they found that 84% of these sites are located in the red area and only 5% of them in the blue area, thus confirming the validity of this hybrid approach that combine statistical analsys with RS data and GIS.

2.1.3 Detection using geophysical methodologies

Some of the methodologies most used by local authorities for detecting illegal landfills belong to the family of the geophysical methods. These methods really represents the true expression of field work in the fight against illegal dumping issue, since they can only be carried out on site with competent personal doing it. Typically these methods require to outline an area and divide it in cells in order to make the whole process more effective in particular in the post acquisition analysis phase. There exist many methods available among this family that can successfully used for detecting wastes and almost all of them are prevalently tuned to actually detect buried waste in the ground rather than detect concentration of waste over the ground. Some possible example are representing by:

- **Boreholes excavations** used to take terrain samples for geochemically analysis
- **Ground Penetrating Radar** (GPR) analysis, used to analyse the terrain on site

Boreholes excavation are widely used since they are effectively reliable in their results just because performing chemical analysis of the terrain is possible to understand if something is buried underground. Some examples of studies that exploited the reliability of boreholes excavations are study [2] that used 50 boreholes excavations as actual ground truth to test their detection RS methods and another study [11] that used boreholes for GPR calibration and for some comparison of the results obtained with the GPR.

Ground Penetrating Radar (GPR) represents another interesting technique that proved to be very successful in finding illegal waste buried underground. A study example that studied the applicability of this technique for the illegal dumping issue is represented by this study [11] that comes from England. In this study the researches pointed out one of the main drawback of boreholes, i.e. the time consuming and costly nature of the technique, and proposed GPR as an alternative faster and less costly approach rather than boreholes. The approach of this study was to demonstrate the feasibility of using GPR to analyse the internal structure and shallow-depth geology of a landfill site, also to identifying possible leachate breakout points in the contaminant wall. The testing area is a 0.4km^2 grid divided in cells, located in eastern England, and inside that area only the cells with the highest probability to be affected by a leachate where examined. The instrumentation is represented by a GPR with a range of antennas covering the following frequencies 50, 100, 200, 225 and 450MHz that was used for two main steps:

- landfill internal structure mapping
- shallow-depth geology mapping

Results from the internal structure mapping showed how the high-frequency GPR dataset coming from the 450MHz antenna is the most suitable for this task and also the feasibility to use GPR to effectively resolve internal ground details making able to distinct the reflections given by the ground, the waste, the leachate and the water. Result from the shallow-depth analysis showed how the high-frequency GPR dataset coming from the 450MHz antenna is the most suitable for this task too pointing out the detection of some distinct sand-gravel lenses in the terrain making clear the feasibility of GPR for this kind of analysis; moreover they studied also the possibility to check weather the containment walls were still intact and results showed how using the 100MHz antenna it is possible to have enough detail to clearly see that there are anomalous feature that can be explained by a breakout point in the containment wall.

All these methodologies are widely used by public authorities that work in the field of illegal landfill detection but still, as precised in some studied, cannot be implemented to large scale because of the costs that will derive from such decision. Moreover these techniques are more focused in the search of buried waste that caused soil contamination rather than waste disposal on the ground. This reasoning however does not mean that they are not useful or do not worth to be used in the scenario they are proposed, the point instead is that these techniques must be part of a greater scenario that sees a large scale work using remote sensing or machine learning methodologies to tell where it really worth to apply these field work techniques to lastly verify the presence of an illegal landfill or even only illegal wastes.

2.2 Deep learning methods for RS and wastes identification

Deep learning (DL) is a sub-family of machine learning (ML) algorithms that are largely based on Artificial Neural Networks (ANNs) that use multiple layers to extract features from data. Deep learning algorithms do not rely to much on strict rules or models fixed a priori but instead exploit the huge amount of data, that nowadays is available, to learn by itself an accurate model that can be used for identification tasks. One class of DL algorithms that got great interest is represented by Convolutional Neural Networks (CNNs) for its great results in image classification[22] in the field of Computer Vision (CV).

Computer vision represents the branch of computer science that works with visual data for tasks of visual identification inside images. The field of CV is a good example in which CNNs proved great results, thank also the huge amount of images easily available online, in tasks like object detection[16], 3D pose and motion estimation[23] for example.
Also the hardware progress played a big role in the interest for CNNs in CV, since the advent of powerful Graphical Processing Units (GPUs), with a huge number of processing cores able to do many small calculations at once, allowed to actually speed up the learning process of the networks up to 40 times, as demonstrated in one the first studies that exploited the usage of GPUs for deep learning [24].
Latest years saw great advancements in the field of CV thanks to CNNs and also great achievements, some good examples are represented by studies like:

- **R-CNN** (Regions with CNN features)[16] that improved detection result by 30% with respect with best previous result on PASCAL VOC 2012 dataset

- **Resnet** that with a very deep architecture won the 1st place on the ILSVRC 2015 classification task[25]

- **Faster-RCNN**[6] that made so efficient and fast the detection process to be almost used in real time scenarios

- **Mask-RCNN**[26] that expanded the detection capabilities of Faster-RCNN to allow the detection of the edges of the objects in the images.

Some of the main tasks that sees constant development in CV, that are interesting for future usages with the dataset proposed by this project, are: image classification, object detection and instance segmentation.

The task of **image classification** (case a. in Figure 2.1) refers to the scenario in which the goal is to learn from many data as possible how to correctly assign one or multiple class labels to an image depending on its contents. The learning process refers to the features extraction process that extracts the necessary differentiating features from the input data that will be used for the classification[27].

The task of **object detection** (case b. in Figure 2.1) refers to the scenario in which other than just classify the objects inside the image they are also localized with a bounding box. An example of approach used for object detection is R-CNN[16] that does: Region of Interest (RoI) extraction, features extraction for each proposal region and classification of each proposed RoI. Possible other algorithms used for object detection are: Faster R-CNN[6], YOLO (You Only Look Once)[7].

The task of **instance segmentation** (case c. in Figure 2.1) represents an advancement that has been done starting from the object detection methods. In object detection methods the model search for the regions that actually can contain objects and then classifies them with the correct class label, instead in instance segmentation the goal is to identify a mask that closely matches the shape of the elements to detect. A possible example of instance segmentation is brought by Mask-RCNN[26], developed starting from Faster R-CNN, that uses two parallel processes one for RoI proposal and one to predict the class also giving as output a pixel-level mask for each RoI found. Instance segmentation actually allows to identify and locate objects giving a more detailed vision of them in images with respect to pure object detection.

There have been different surveys that summarize how Deep Learning was

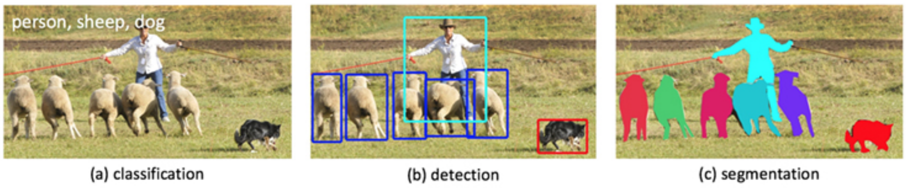

Figure 2.1: Computer vision tasks[28].

applied to remote sensing in the last years [29, 30, 31]. Works in the area have gone from the identification identification of ships using RS imagery [32] to the identification of landforms, such as the case of mountain peaks using digital elevation models of the terrain [33] or the case of craters in the moon or mars [34, 35]. Another example is the work [36] that has demonstrated the possibility to apply DL algorithms to effectively exploit RS imagery proposing a DL approach to automatically detect terrain features. For this study authors manually collected and labeled more than 100 of RS images, with 1m and even higher resolution, to create a dataset of terrain features according to the characteristics that the features need to have: definite boundaries, non small area and non-vertical characteristic. Results obtained on the dataset they created showed an overall accuracy of 91% in detecting terrain features. This study confirmed the real possibility to profitably use RS imagery along with DL approaches.

With reference to the illegal landfills scenario, an interesting study that pointed out the feasibility of using DL methodologies to improve already existing approaches of human visual detection, towards more autonomous ones, is represented by this study [8] that comes from San Jose, California. This study proposes the use of a fully automated deep learning approach for wastes detection based on images that comes from intelligent cities surveillance camera-based monitoring systems. These monitoring systems was placed in specifics areas that represent the locations with the highest probability of finding illegal dumping situations. The solution proposed by this study comprise a first step of identification where the most frequent disposal wastes have been chosen among hundreds of possible choices, a second step in which they acquired from local authorities information about the hottest spot in

the city of San Jose where is very probable to find illegal dumping and last step they predisposed an architecture made with a server and many edge computing stations (installed in the individual hot spots). Edge computing stations have the objective to avoid server overloading by acting as filter only sending to the server the images that have very high confidence to contain the targeted waste.
They implemented both AlexNet and GoogLeNet model trained with 1423 cropped images, containing only the class elements to detect. The classes of wastes considered were: mattresses, sofa, furniture, trash, electronics, carts, trash bag, tree.
The results that this study obtained demonstrated interesting accuracy results, that favored GoogLeNet compared to AlexNet showing also that a deeper architecture in conjunction with more training iteration is successful. Except for carts class, that performs poorly in both models, AlexNet accuracy results are around the 50%, instead GoogLeNet results show an accuracy of near 75% for the classes of electronics, mattress, trash bags; an accuracy of near 90% for the remaining classes. This study concretely demonstrated that deep learning methods can be successfully used also for waste disposal detection really improving all the currently known manual approaches that in many cases required an human intervention for the whole process of detection making it costly and time consuming.

2.2.1 Deep learning datasets

When considering deep learning scenarios having a well structured dataset is as much important as having a good model suitable for the context in which it will work. A well done and structured dataset to work with thus can be the key to train model to successfully classify or detect objects. Keeping this in mind studies like [9][10] proposed the importance following certain rules that can help in creating greater quality datasets that are better organized and allow better accuracy. An example of incredibly well done hierarchical dataset is the case of ImageNet[9] that is structured to have specific properties that ensure the dataset quality. These properties are represented by:

- **scale**: represents an important factor to consider since it allows the have a more generalized dataset with many different categories available
- **hierarchy**: offers a good way to implement a very dense dataset rich of subcategories easily scalable
- **accuracy**: is a very important property since it actually represents a measure of how well the images describe the categories they belong, as the authors pointed out in case of hierarchical datasets keep a very high accuracy also in deepest levels of hierarchy is not easy since elements in those levels are very hard to recognize
- **diversity**: inside a category objects should have variable position, appearance, poses, view point, background clutter and occlusion. This helps a lot in creating more generalized categories inside the dataset that will end in describing them more accurately

To allow the development of DL approaches able to exploit RS imagery, specific studies voted to the realization of GIS specific datasets have started in the latest years like DOTA(Dataset for Object deTection in Aerial images)[37] and iSAID(Instance Segmentation in Aerial Images Dataset)[18].

The motivation behind the DOTA dataset creation lies in the fact that object detection in aerial images has different needs mainly due to these issues:

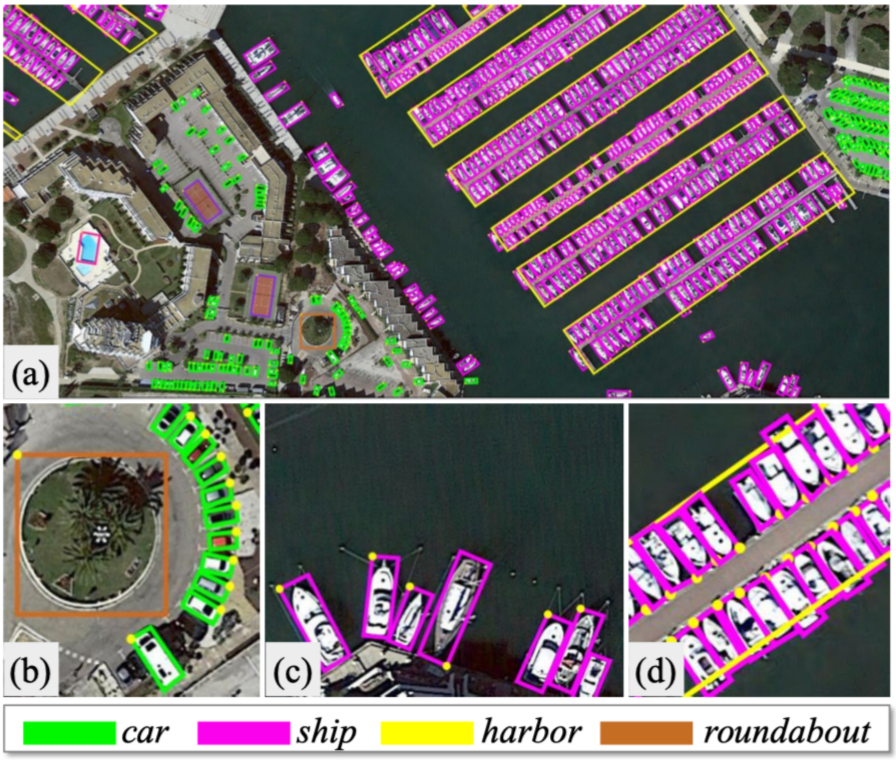

Figure 2.2: Annotations example from DOTA[37].

- The scale variations of objects in aerial images are rather big compared to the ones in normal images
- The presence of many small objects in crowded disposition
- The presence of objects often with different orientations

Keeping these issues in mind authors developed DOTA dataset from 2806 aerial images, each of them with size up to 4000×4000 pixels, that contains objects of different scales, orientations and shapes. These images have been annotated by experts in aerial image interpretation, with respect to 15 common object categories. The dataset thus contains 188282 instances of bounding box annotations[37], whose an example of these annotations can be seen in Figure 2.2.

A further improvement of DOTA, always in the field of dataset tuned from aerial imagery, is represented by iSAID. iSAID is a dataset meant for instance segmentation upon aerial imagery built with the same criteria that allowed DOTA to be optimized for object detection in aerial images and developed starting from the same images used for DOTA. iSAID is constituted of 655,451 instances of segmentation masks built upon 15 classes, different from the ones of DOTA.

The realization of DOTA and iSAID, highlighted the necessity of building specific dataset when the identification task has the interest to be tuned for the usage of RS imagery, proving that normal datasets do not necessarily reflect the complexity that is hidden inside aerial and satellite images.

These two studies are not the only one that in the latest years exploited RS imagery for the creation of DL datasets.

This study[38] brought to the creation of xView, a dataset for object detection based on satellite imagery from WorldWide-3 satellites with 0.3m of ground sample distance. The xView dataset contains over 1 billion of detected objects divided in 60 classes for a total spatial covered area of more than 1400km^2.

Another study that exploited satellite imagery for the creation of a dataset is represented by study[39] that produced BigEarthNet for RS images understanding. BigEarthNet is a multi-label dataset for image classification that consists of 590326 images from Sentinel-2 satellite (using the bands of 20m and 60m resolution) with more than 40 class labels considered to describe the images.

In the attempt of realizing datasets from RS imagery of even greater quality it came up the necessity to tune them towards real worlds application scenarios to alleviate the bias problem that comes from a potential low degree of generalization of such RS datasets, as pointed out in[37].

There are therefore several studies that produced dataset more focused on a specific issues.

An interesting example is case study[40] which proposed a dataset for segmentation tasks, based on satellite imagery, for the assessment of changes and building damage detection of an area in correspondence of natural disasters. The xBD project collected multi-temporal images of areas affected by a natural disaster to have pre- and post-crisis imagery of the areas. In this way they made possible to quantify the damage done to an area from the natural disaster with four different levels of damage: no damage, minor damage, major damage and destroyed. From this study arose possible usages[41] of the dataset for the identification of new road routes after a disaster, necessity due to the creation of obstructed roads, or for the effective and better identification of the force of nature involved in a specific disaster, allowing responders in charge to make better decisions to reduce further damages.

A different example is case study[42] that brought to the realization of another interesting dataset, called Agriculture-Vision, that makes use of satellite imagery to detect agricultural patterns. Agriculture-Vision is a dataset for semantic segmentation created from RGB and NIR images of very high resolution (0.1m) annotated with the scope to detect nine types of field anomaly patterns that represent an issue for farmers. A few examples of these patterns are represented by: planter skip (areas in which seed are missing), storm damage and dry down of an area.

The results from all these studies motivated us in creating a dataset of RS imagery specifically annotated for the identification of illegal landfills.

2.2.2 Tools for annotations

The tools used for the creation of datasets represent a crucial part of the creation process. Datasets needs to be large to be enough generalized and this requirement means that the creation of a good dataset necessitate to annotate thousands of different images, process that is very time consuming.

To overcome this issue also researcher teams that developed very large datasets had to find a solution to ease this long process. For example for the creation of ImageNet and MS COCO the author have exploit the solution offered by Amazon Mechanical Turk (AMT)[1]. AMT is a web service, from Amazon, that allows requesters to propose tasks that many interested web workers can complete and for which they will be payed. There exists other options like tools developed to allow and ease the annotation of images, with many different characteristics. Some examples of open-source tools can be:

- **OpenLabeler**[2]: a standalone tool developed as desktop application
- **VGG Image Annotator**[3]: a browser based standalone annotator
- **coco-annotator**[4]: a browser based standalone annotator
- **ImageTagger**[5]: an open source online platform for collaborative image labeling

However such tools did not have the right characteristics needed for the creation of the dataset that this project wants to achieve, in particular because they do not work with georeferenced data. There have been other works that implemented crowdsourcing to annotate geolocated elements, for example in [43] they propose the analysis of aerial imagery for land cover classification, in this case, the tool has some fixed question regarding the use case, another example is [44] in which the authors propose a tool to annotate

[1]https://www.mturk.com
[2]https://github.com/kinhong/OpenLabeler
[3]https://gitlab.com/vgg/via
[4]https://github.com/jsbroks/coco-annotator
[5]https://github.com/bit-bots/imagetagger

landforms directly on a 2D and 3D map, such tool presents markers of possible locations where the users need to accept or reject the proposal. Such examples, among others [45] are not useful in our scenario, since they do not provide the option to draw polygons to indicate the specific features we are interested in, moreover in some cases the data asked to the user is highly tuned to their scenario.

To the best of our knowledge, none of the tools includes the right characteristics needed for the creation of the dataset needed, namely: consideration of georeferenced entities, among with the possibility to use our source of preference, the web-accessibility to allow cooperative work among workers and the protection of the data to render it accessible only to some users.

For this reason we decided to create a custom web annotation tool with georeferenced images support and the possibility to allow the cooperative work to speed up the annotation process; moreover a custom tool allows also to tune it to particular needs, for example the possibility in the future to import DL result and directly showing it on the images.

Chapter 3

Dataset

This chapter will cover how and why the suspicious sites lists and imagery sources have been chosen and the actual locations used as observation areas of study taken into account for this project. Moreover this chapter will also cover the strategy used to make the annotations and the labels took in consideration.

3.1 Data sources

The main sources taken in consideration are satellite images and aerial orthophoto images, both these sources provide colored images in the visible light spectrum. The aerial orthophoto images are airborne imagery that have been corrected in order to remove optical distortions. The choice of these two typologies of images relies in their high resolution allowing to see in the images also very small details of less than 5 meters; with reference to satellite images another reason that made them worthwhile for this project is the fact that they are easy to retrieve and thus a good starting point to work with. The possibility to have such high resolution images allows to spot every kind of small wastes typical of small illegal landfills and not only of the greatest ones overall increasing the detection range capability available.

The point of this dataset was to create something useful for detection not only of illegal landfills by them self, thus depending on their shapes or intrinsic features, but also of the wastes that characterize an illegal landfill. This has been necessary because from visual inspection of many illegal landfills already detected it came clear that, even if from a features point of view there are common features to many illegal landfills, not always these common features are easy to recognize as a visual pattern. There exists great diversity in the shapes and features of the possible illegal landfills that can be found. For this reason an approach that differentiate among all the types of wastes commonly found inside illegal landfills allows to easier spot them. This decision also allows to spot small wastes disposals that maybe cannot be yet considered as illegal landfill, because of their small sized or because still in an early phase that could potentially evolve in a new illegal landfill, thus being worthwhile to be studied. Taking this wastes differentiating approach has allowed to create a multi-class dataset of wastes that will be used in the future to train deep neural networks for the detection of illegal landfills and their constituting wastes.

3.1.1 Suspicious locations in area of study

The whole realization of the dataset started from already prepared lists of existing suspicious illegal landfill sites. Part of these lists have been found on the internet from regional and/or provincial websites who published them with cooperation of local authorities. These lists typically contained: an ID number, an identification name, often but not always the coordinates of the location (some lists used WGS84 CRS and some other UTM zones 32N or 33N CRS) or in some other cases just the national address of the location (thus requiring to geocode it to actually acquire the coordinates for that location), in some cases a description of the type of wastes and the severity of the landfill found, finally in certain lists was also available an information about the status of the landfill (if the environmental remediation and recovery is not yet started, in progress or completed). The lists found online were not always comprehensive of all the necessary data, in these cases a cleaning step was required; an example of these cases is represented by the lack of the minimum information needed by the geocoder to retrieve the correct

coordinates of the site. Some lists with missing coordinates or addresses too broad to be correctly geocoded were thus discarded in the process. Another issue that had to be considered is that elements from the lists are taken in a longer time span sometimes increasing the difficulty of locating the site. It is possible that it has been already reclaimed or no more active at the time of the observation.
The lists that were found to be of good quality and thus used in this project cover large areas of Campania[1], Piemonte[2] and Lombardia[3] as can be seen in Table 3.1.

Region	Covered Area	Web Source
Campania	Many provinces of Campania region	ARPA Campania
Piemonte	Many provinces of Piemonte region	Geoportale Piemonte
Lombardia	Metropolitan area of Milano and nearby	Open-datahub sciamalab

Table 3.1: Covered areas from online found lists

Another part of the data sources that have been used for this project are provided us by ARPA Lombardia (Agenzia regionale per la protezione ambientale) that shared with us their know-how on the illegal landfills issue, their knowledge on the wastes that most describe an illegal landfill and the aerial orthophoto imagery of a large areas of Lombardia. These suspicious sites lists contain similar information to the ones publicly found plus they contain also the evaluations done by ARPA Lombardia about the size and the severity of the considered sites. The information of these suspicious sites lists seem to have greater quality in comparison with the ones publicly

[1]http://www.arpacampania.it/web/guest/1408

[2]http://www.datigeo-piem-download.it/direct/Geoportale/RegionePiemonte /Ambiente/ASCO_Anagrafe_Siti_Contaminati.zip

[3]https://www.sciamlab.com/opendatahub/dataset/r_lombar_x774-7qxt

found online, plus have been created starting from the same aerial orthophoto imagery that ARPA Lombardia gave us for this project making even more unlikely the scenario of not locating the sites contained in the lists inside the images. The sites of ARPA Lombardia are specific of three provinces of Lombardia that can be seen in Table 3.2.

List	Covered Area
Pavia	Pavia's province, specifically many municipalities inside the area called "Lomellina" of its province
Lodi	Lodi's province, specifically many municipalities inside its province
Brescia	Brescia's province, specifically many nearby small municipalities around Brescia

Table 3.2: Covered areas from private ARPA Lombardia lists

From the visual inspection of the suspicious sites images from the different regions it is possible to understand that not all the territories are similar to the others. There are territories more characterized by countryside and rural areas while others by urban and suburban areas and others hybrid that are a combination of the previous two types. Inside countryside and rural areas sites can be found in open fields, near farmhouses, hidden in proximity of woods and typically are represented by scattered wastes stacked up in most hidden way. Inside urban and suburban areas sites can be found near ruined buildings and often also near old or closed construction yard and industrial sites; in this scenario there are many more types of wastes that can be found typical of the urban or industrial areas like pallets, IBC, wreckage and old tires. For this specific reason, in order to have a dataset capable to describe in the best and most generalized way all the possible types of illegal landfills, we decided to choose areas that from a territorial point of view complements each others very well thus obtaining a collection of suspicious illegal landfills that is the most various possible. In Table 3.3 we shortly describe the characteristics of each area.

List	Territorial typology
Campania area online source	mainly countryside and rural areas and also suburban areas nearby many provinces of the region
Piemonte area online source	both countryside and suburban areas in many provinces of the region
Milano area online source	mainly urban and suburban areas around the metropolitan area of Milano
Pavia area ARPA source	mostly suburban areas with some countryside and rural areas in the province of Pavia in particular in the are called "Lomellina"
Lodi area ARPA source	both industrial, suburban areas and countryside areas in all the province of Lodi
Brescia area ARPA source	both urban and industrial areas around the city of Brescia

Table 3.3: Chosen lists with their territorial description

3.1.2 Mapbox satellite imagery

The main imagery source used to retrieve satellite imagery is the map provider Mapbox[4], that we chose since it was easy to use with great and well documented API and also with the possibility to use the images both offline for deep learning tasks and online for the purpose of annotating the images. These represent the main reasons why Mapbox has been chosen as main satellite imagery source for this project.

Other satellite imagery source providers have been tested along with Mapbox, e.g. Google and Bing from Microsoft, but they all shared some restrictions, due to the terms of use, that for the purposes of this project were actually very problematic, in particular: impossibility to permanently download the images for offline uses, impossibility to remove watermark directly

[4]https://www.mapbox.com

from the API and impossibility to use them for derivative works that get results from their data.

Mapbox provides very powerful and complete APIs to access its data maps that allows to choose between different kind of style maps/imagery and different formats of download.

Mapbox APIs allows many possible choices for the style of the maps, available types comprehend: satellite view, streets view with only stylized color of terrain, satellite view with also street layer applied and many others. The style of choice for this project is pure satellite images without any other layer applied upon them.

Images in Mapbox APIs can be requested in many formats and the two interesting for the goals of this project are Raster Tiles and Static Images, but before going any further is important to understand the difference that exists between them.

Mapbox Raster Tiles API[5] returns on tile per request and takes in input the tile coordinates, the zoom level, styles ID, high-density display parameter and the access token needed to authenticate the request and returns the requested tile in jpeg format.

A tile represents one image patch inside of a whole greater map, actually represented by the whole Earth, for a specified zoom level. As the zoom increases the whole Earth, represented as a rectangle, is progressively cut in squared tiles that have their tile coordinates represented as the column and row indexes of that tile inside the greater rectangle that contains all the tile for that zoom level. The same couple of coordinates at a certain zoom does not corresponds to the same location at another zoom level, this happens because the rectangle representing the Earth is cut in a different number of tiles at each zoom level. As the ultimate consequence of this each zoom represents a complete dataset of tiles (a tileset) that represents the whole Earth at a certain zoom level. Since the tile size, in term of pixels is always constant, the zoom level represents a measure of how much area is shown in a specific tile.

[5]https://docs.mapbox.com/api/maps/#raster-tiles

This is the typical system used for example by map plugins in web applications, like Leaflet[6], and is commonly referred to as Slippy maps.
The whole point of using this API is actually download very small patches of images at high zoom (the raster tile in this case) and then create a mosaic image using this patches, doing so the final high quality image allows to see every possible detail inside of it.
Each tile is a 512x512 pixels image that, at the maximum zoom, covers a squared area of ≈150m of edge thus with a resolution of ≈0.3m per pixel. The borders coordinates of the tiles can be easily computed and this is very important when creating images from raster tiles because every tile covers a different area depending on its latitude.

Mapbox Static Images API[7] instead returns an image that is the result of tiles already summed together and rendered by Mapbox; for this API the requested inputs are: coordinates of the image center, zoom, bearing (rotation with respect to the north), pitch (angle at which the image is taken), resolution of the final image, high-density display parameter and access token to authenticate the request. This API is really useful because is able to provide directly a complete image, that in this case scenario will represent a potential illegal landfill to examine, with only one request. For example this allows it to be used in an online scenario, with loading time of near 1 second for an image of 2560x2560 pixels.

However, from a direct visual comparison of images derived from these two APIs, the images created as a mosaic of raster tiles have greater quality and more sharp edges on the elements contained, probably due to slightly compression issue in static images, making them more suitable for deep learning tasks.

3.1.3 Orthophoto aerial imagery

Orthophoto imagery is the other imagery source used in this project, in this case images are taken as aerial photography but what makes them really

[6] https://leafletjs.com/
[7] https://docs.mapbox.com/api/maps/#static-images

different from normal aerial images in terms of quality is the way they are prepared since these images are geometrically corrected, thus it is important to understand what is an orthophoto and how orthophotography works before continuing. An orthophoto is an image taken from altitude but actually that kind of image without any preprocessing suffers from distortions, e.g. projection distortions of different planes at different elevations, lens/sensor distortions and camera angles distortions. To avoid this issue the orthophoto is geometrically corrected through an ortho-rectification process that corrects these defects so that the image can be used for mapping purposes and for meaningful distances measurements. For this reason ortho-rectification process is very important, even more considering that the higher it is the resolution of the image the higher the defects will be strong in the image.

The orthophoto images that this project uses have been granted by ARPA Lombardia that has collaborated with this project not only with the imagery but also with the lists of potential illegal sites. This orthophoto imagery have been provided in ECW (Enhanced Compressed Wavelets), a compressed format used for images that are used in GIS, and each image actually represents a very large area of $\approx$10km^2 with a resolution of $\approx$0.2m per pixel. The coverage of these orthophoto images is of three provinces: Pavia, Lodi and Brescia. The Coordinates Reference System (CRS) of the original compressed ECW images is not WGS84 but instead Gauss Boaga Western Zone, also referred as Monte Mario/Italy zone 1 or EPSG:3003, that is a local coordinates reference system used in Italy as a standard until some years ago when it has been superseded by UTM zones standard. This CS is specifics for the coordinates of the western part of the Italy, for example coordinates inside Piemonte and Lombardia are part of this set. For this reason it has been necessary to convert the coordinates in WGS84 to make them usable in more generics settings. WGS84 stands for World Geodetic System revision 84 and is the CRS that uses latitude and longitude coordinates expressed in degree. Valid range for latitude is -90° to +90°, while for longitude is -180° to +180°.

3.2 Annotation strategy

The annotation strategy adopted for this project has the primary goal to produce a dataset for image classification, object detection and instance segmentation tasks in computer vision.

A dataset able to meet these requirements has to be multi-class: a class that represents the whole illegal landfill and multiple classes that accurately describe the waste types that are typically contained inside of it.

The choice of multi-class dataset comes also from the fact that not always it is easy to detect a potential illegal site in its entirety. This issue happens because illegal sites can have great variety of shapes and typologies and also because there exist the possibility of finding such small concentrations of a single wastes typology (for example many pallets in a 5x5m area behind a building) that cannot be classified alone as an entire site but still are important to be detected.

A correctly annotated image is represented by an image in which the illegal landfill site and all the waste elements have been annotated with valid annotations.

A valid annotation, in order to be considered as such for this dataset, must be made by a polygon that outlines the shape, closely matching it, and must have a class label, to describe it, referred to the element annotated.

When an illegal landfill is found also the elements strictly contained in it must be annotated, as final result its annotation will enclosure the annotations referred to the wastes. It is always preferable find the illegal landfill that enclosures other annotated waste, although this annotation strategy takes also into account the scenario in which an illegal landfill is not clearly recognizable in the image while instead other wastes are easily found. In this case annotating only the other wastes, with their appropriate class labels, will still produce a correctly annotated image.

In Figure 3.1 an example of correctly annotated image is presented, inside of it is possible to see what is the actual output of our annotation strategy.

A straight line polygon is used to indicate the valid polygon annotations that we expect for each element, with reference to the colors of the example green is to indicate the illegal landfill and the other colors, namely red and blue, for the waste elements.

A dashed line is used to indicate the Bounding Box (BB) that is inferred from the valid polygon annotation.

Figure 3.1: Example of an image correctly annotated.

Potential Deep Learning usages

With reference to the deep learning algorithms that are used in the field of computer vision, from the annotations created with this strategy and this format it is possible to extract different type of knowledge. As shown in Figure 3.1 the result of annotation process on an image is a set of annotations that accurately outlines the shape of the illegal landfill and its waste with attached their class labels (straight line annotations). From these annotations it is possible to perform instance segmentation tasks aimed to find, outlining their shape, the wastes in images classifying it with their correct class. From these annotations have been computed the BB (dashed line rectangular annotations) that can used to perform object detection tasks aimed to locate with a BB the wastes in images classifying it with their correct class.

Instance segmentation and object detection tasks are not the only possibilities available actually, even though the annotations are primarily meant for those tasks, it possible to extract the class features only from the annotations realizing a datasets of only labeled images, e.g. the example in Figure 3.1 would have the labels of the illegal landfill and of the specific wastes contained in it. From a set of images labeled in this way it is possible to perform image classification tasks aimed to tell which classes of waste describe better the image.

3.2.1 Class labels

The choice of all the class labels was initially based on the knowledge that ARPA Lombardia shared with this project, which provided a great insight on what are the elements that can effectively describe an illegal landfill. This starting knowledge was further refined with the experience gained during the annotation process. The class labels used for this project are:

- **generic site**: class label that represents the potential illegal landfill which is meant also to encapsulate inside the other class labels
- **scattered wastes**: wastes typically composed of scattered trash, rubble and wreckage
- **pallets**: a flat transport structure, which supports goods in a stable fashion while being lifted forklift, typically made of wood but also in plastic sometimes is easy to find in areas where wastes are amassed
- **intermediate bulk containers (IBC)**: used for containment of liquids, this represents a very dangerous waste since often the liquids contained can be toxic and an inadequate storage can damage the IBC causing it to lose liquids in the ground
- **dumpsters**: like small containers opened on top, typically full of every kind of wastes, can be commonly found in groups inside illegal landfill sites
- **containers**: big containers closed on top like the one used for road, railway or naval transportation, like dumpster can be found in groups inside illegal landfills
- **tires**: abandoned tires made of plastic or rubbery material that deteriorating can be dangerous for the environment
- **plastics bags**: in some illegal landfills it has been found that waste are actually enveloped around plastic bags, probably to help moving it around the area or to hide them more easily

- **tubes**: metallic or plastic tubes for building use that are typical of construction yards that have been abandoned, these wastes represents a potential threat for the environment since when left to prolonged sun exposure or bad weather can deteriorate releasing toxic wastes in the ground

- **hay bales**: in more rural areas can be commonly found in very large numbers outside their right storage place, sometimes near woods where they can easily hid

- **wood**: typical of rural areas there is a limit in the quantity of material that can be stocked on the soil in an open area without any specific permission to do it, actually this class refers to the case there is a very large amount of wood placed on the ground

- **generic wastes**: compared to scattered wastes this class is representative of wastes that are more structured in shape and that cannot fall in the other class; e.g. domestic appliances

All these class labels are representative of wastes elements that characterise an illegal landfill but of course are not only wastes elements that can be found inside it, for this reason we decided to add the generic waste class.

These particular classes were chosen among all the possible choices because these are actually the easiest elements recognizable from satellite images and also the ones that are often found in the same fashion inside the images.
In Figure 3.2 it is possible to see one example of image per class label.

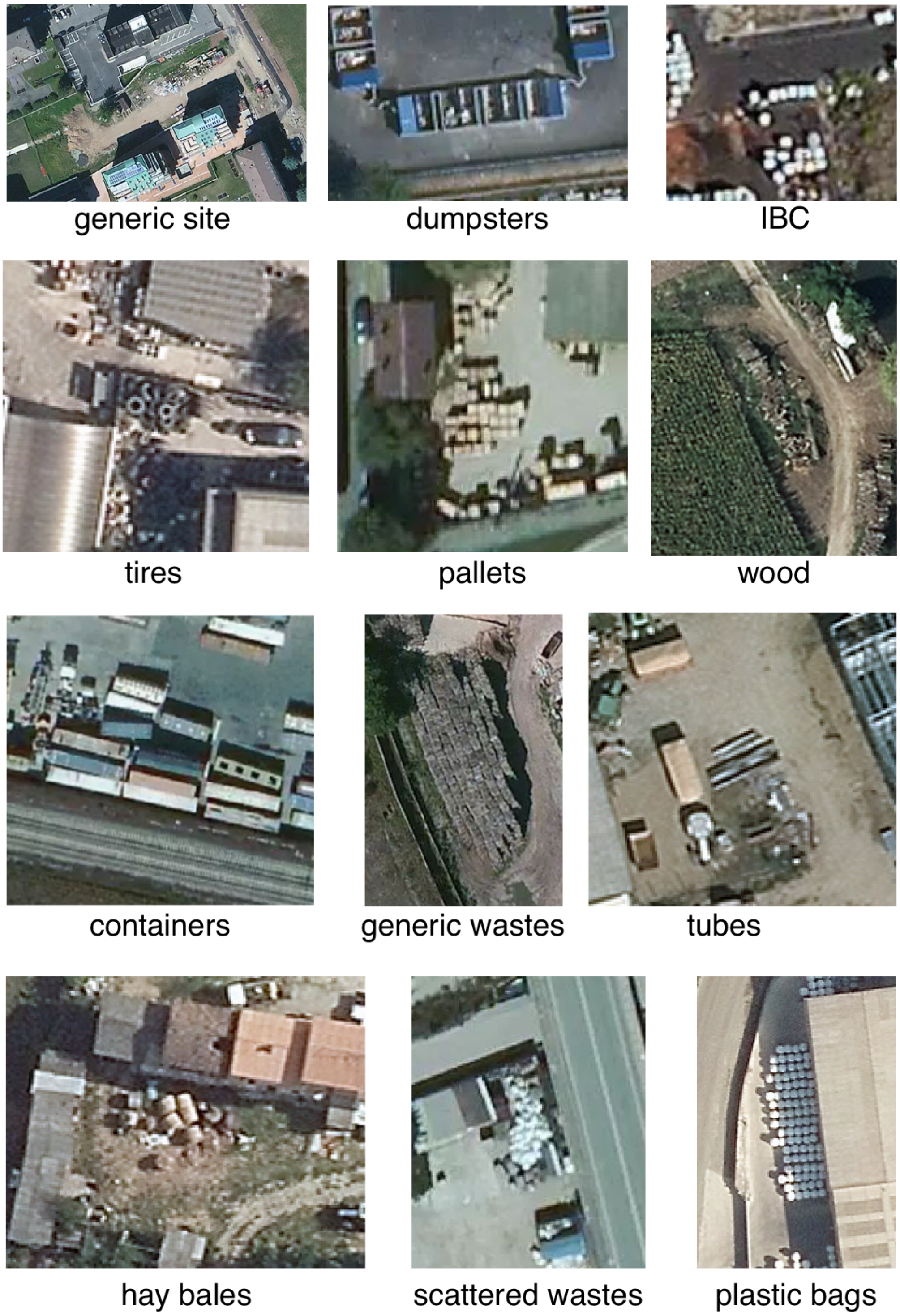

Figure 3.2: Example of the class labels.

Chapter 4

Tools

To create the dataset explained in Chapter 3 we needed specific tools to retrieve all the needs images, process and prepare them to be finally annotated. In this chapter we introduce the crawler used to retrieve the satellite imagery from Mapbox provider, the orthophoto service that pre-processes the original orthophoto images and make them available through a web service and all the requirements and implementation details of the web annotation tool used to effectively create the annotations that compose the dataset.

4.1 Mapbox crawler

In order to be able to realize the dataset it is important to obtain all the necessary images that have to be annotated in the web annotation tool, which will be presented later in section 4.3. From this need comes the necessities of having a tool for gathering high quality images to be used offline to effectively create the dataset. For this reason a crawler has been developed using python[1] language. This way we are able to download images while saving their associated geographical metadata. The crawler contains two sub-modules one for each Mapbox API (static images and raster tiles) described with more details in previous sections (Section 3.1.2).

[1] https://www.python.org

The two sub-modules available, with their respective APIs, are:

- **Static sub-module**: referred to Mapbox Static Images API
- **Tiles sub-module**: referred to Mapbox Raster Tiles API

Both the sub-modules of the crawler, to download the images, operates using as main inputs:

- **ID** that gives a unique identifier to the image that will be saved
- **Latitude** coordinate of the center point of the image
- **Longitude** coordinate of the center point of the image
- **Distance** that represents the edge distance measure of the final image

The Static sub-module of the crawler downloads one full image per request. In this API the area of the final image depends only on the zoom and the resolution parameters specified in the request. The resolution parameters in our scenario is always fixed to maximum available that is 2560x2560 pixels, thus allowing to get best quality possible in the final image. Since the zoom parameter is the only one that can be used to choose the area of the final image the crawler computes the right zoom to make a request that returns an image that covers the area specified by the distance parameter. The only minor issue with this process is that the zoom value given in the request to API has a finite precision of two decimal after the comma and this results in a little imprecision in the edge distance represented in the image, this imprecision is estimated around ≈0.4-0.6% that in an image of 500m of edge is actually ≈2-3m. To overcome this technical issue also the Tiles sub-module of the crawler has been modified to take in consideration this imprecision in the edge coordinates. The Static sub-module of the crawler actually downloads the same exact image that is available to the workers for annotations in the web annotation tool.

To test if there was the possibility to download from Mapbox images with even greater quality, assembling them from raster tiles, it has been decided

to develop another sub-module of the crawler to work with Mapbox Raster Tiles API.
The Tiles sub-module of the crawler that downloads images in different operating modes:

- **distance**: downloads a squared image that has as edge size(in km) the exact distance
- **distance with resize**: operates like distance mode but in addition it resizes the image to a specified resolution as last step
- **pixel**: downloads an image depending on the width and height given in input
- **distance with coordinates correction**: operates like distance mode but in addition it corrects the geographical border of the final image to be equal to the one downloaded from static images API

The main process of this sub-module, in addition to the standard inputs, receives as input the zoom that needs to be used for the tiles that will compose the image and depending on the operating the scale resize parameter or the width and height that the final image has to meet (pixel mode).

The zoom parameter is important in this sub-modules of the crawler since it is actually more referred to the quality of the final images rather than to their area represented. This happens because the zoom refers to the area represented by a single tile. For each zoom exists an entire tileset of images that represents the Earth with at a certain image resolution. Thus creating a mosaic image with a lower zoom parameter requires more tiles (each of them at low resolution representing a large area), instead using an higher zoom parameter requires more tiles (each of them at high resolution representing a small area). The zoom parameter, for our purposes, is fixed at the maximum level of quality available for the area of study considered, allowing to create images of the highest quality possible.

Once received the inputs the crawler starts computing the tiles needed for a particular image with that particular edge distance and then downloads them. To computes all the tiles needed the crawler starts finding the tile coordinates of the tile the contains the center coordinates of the image (the ones that are given in input), then it finds the tiles that contain the corners of the final image and finally, with those information, computes the list of all the necessary tiles.

After all the tiles needed are available it starts the merge process of the tiles that will compose the final image, at that point there is a step in which edge coordinates of the image, resulting from the merge of the tiles, are computed. This is needed to crop the image to the coordinates of the desired area, at that point the image is ready to be saved.

Both sub-modules of the crawler can work receiving a list of coordinates from an input file in csv (Comma Separated Values) format thus automatizing and easing the download process of large batches of images.

The presence of the Static sub-module was maintained, even though the images created from the tiles are slightly better, because the images downloaded from this sub-module are exactly the same in every pixels to the one that are shown in the web annotation tool that will be later described in this chapter.

4.2 Orthophoto service

Since the aerial orthophoto images are compressed, represent large areas and are in a non standard CRS it arose the necessity to develop a preprocessing tool able to actually create small images starting from the original orthophoto images. Moreover to make the processed images available for the annotation process it was needed also to create a web service capable of returning a specific image upon a specific request (including latitude, longitude and edge distance of the final image). For this reasons have been developed a preprocessing tool and also a web service which constitute the Orthophoto Service.

The orthophoto preprocessing tool has been designed with two objectives in mind:

- creating a tool to cut small images, in an easy to open format, from the original bigger and compressed ones
- creating a modular tool that could be called from a web service to serve online the processed orthophoto images

For these exact reasons the tool, written in python language, was realized in a modular fashion that comprehends the following steps:

1. **metadata extraction** from ECW
2. **map coverage creation** from ECW metadata
3. **computation of necessary ECW** files needed for the creation of a specific image depending on the inputs
4. **conversion of orthophoto** from ECW to GTiff (Geolocalized Tiff images)
5. final **image creation** process

All the steps that actually had to work on ECW files where executed using GDAL[2], a translator library for raster and vector geospatial data formats that is released under an open source license by the Open Source Geospatial Foundation.

The process starts from the metadata extraction in order to understand, for each ECW image, the area that represents. This extraction process is executed on all the ECW images using the gdal_info tool that is able to read the metadata contents without having to convert the whole image; after the extraction the data are parsed and converted in JSON format in order to be easily accessed for other purposes. Coordinates inside of metadata are actually in Gauss Boaga Western Zone format and thus later for the creation of the overall map coverage geojson will require to be converted in WGS84 format.

After this step is completed the whole coverage map is computed and encoded as a geojson that contains a feature for each ECW source image

[2]https://gdal.org/

reading the coordinates from the extracted metadata and converting the Gauss Boaga Western Zone coordinates into WGS84 coordinates; the choice of a geojson format has been done to exploit Shapely[3], a library, that among its functions, checks if a point is contained in a geojson feature.

Once the map coverage have been computed the border coordinates that the final image will have are checked inside the map coverage. This step finds which ECW source file are needed for the image creation, in this way the output will be a list of ECW source files that represent the south-west, south-east, north-east and north-west edge coordinates of the final image. There are 4 possible cases:

1. all corners inside the same source
2. two corners up inside one source and two corners down inside another source
3. two corners to the left inside one source and two corners to the right inside another source
4. all corners inside different sources

Since it is possible that the ECW source files are partially overlapping each others it happens that for a certain coordinate are returned multiple ECW source files, in this case scenario the algorithm always optimizes the output trying to choose the ECW that brings to the easiest case scenario possible, e.g. when possible Case #1 that does not require any sum among multiple images source. This optimization not only improves the quality of the final images, since it can happen that one column/row of pixels is redundant in the sum process (this can be caused by quantization errors done when the ECW have been created and the geographical metadata have been applied to it), but also greatly improves the execution time since it will require far less conversions and GTiff images openings. In Figure 4.1 it is possible to see an example of Case #4.

[3] https://pypi.org/project/Shapely

Figure 4.1: Example of Case #4. Pink lines represent the separation between ECW images, while red square represents the final image that will be computed.

When all the sources needed to compute the final image have been found it starts the conversion process that uses the gdal_translate command to convert the original ECW in an uncompressed GTiff images. The necessity to convert only ECW needed to create the final images arises from the constraint of not having to much permanent space to keep allocated all the GTiff images already converted. To make an example an ECW in our image pool has an average size of ≈200MB and once decompressed in GTiff can have a size of ≈700-800MB, a full conversion of our pool would require just under 1TB space. In future optimization of this process it will be considered to have all the ECWs already converted in GTiffs thus avoiding the need of this conversion step.

From the GTiff images it starts the processing of the final image, following the same cases listed before, that computes the cuts needed for the final image. Depending on the case scenario the image processor cuts from every image the part needed and, in cases different from Case #1, it sums them all. In any case the reconstruction of the corner coordinates of the final image is performed.

Rasterio[4] has been chosen as main library to work with during the image processing, since is a very powerful imaging library that makes really easy to work with GTiff images. To be able to exploit Rasterio's capability to directly read the coordinates inside the GTiff coordinates matrix (which couples together a specific pixel to a specific coordinate) it has been necessary to use Gauss Boaga Western Zone coordinates during the whole image processing. In this way it has been possible also to avoid any kind of projection error due to the differences between WGS84 and Gauss Boaga Western Zone. After all the coordinates computation are completed and the image is completed too then both Gauss Boaga Western Zone and WGS84 border coordinates are saved, thus allowing to have precise geographical metadata about the created image.

Each of these steps is actually represented by a module that can be used by itself, thus allowing to use of the preprocessing tool for the creation of images from a list of coordinates or even attached to the web service that is called by web application annotation tool to compute images that are not yet computed on the server.

Since the orthophoto preprocessing tool has been written in python language we chose Flask[5], a lightweight WSGI (Web Server Gateway Interface) web application framework, to create the web service that connects to it. This web service accepts requests, containing coordinates and distance of the desired image, and returns a previously computed image, if found, otherwise calls the preprocessing tool to compute the requested image and persist it on disk for future requests.

[4] https://github.com/mapbox/rasterio

[5] https://palletsprojects.com/p/flask/

4.3 Tagger tool

In order to fulfill the main purpose of creating datasets of annotated and classified images we found necessary to develop a web application able to do this. We created a specific tool since the other existing ones do not take into consideration the georeferenced nature of the problem our project. For example our tool consents to explore the site locations of interest exploiting interactive maps to show them, or to create and export annotations that are already georeferenced. The point of creating such georeferenced annotation tool is to allow, cooperatively with many users on the same task, the inspection of interesting sites, inserted inside of campaigns uploaded in the web application, in search for objects of interest to be annotated and classified, depending on the specific scenario of that interest. For example in the context of this project has been utilized to create campaigns containing suspicious sites, that needed to be inspected in search of potential illegal landfills and wastes to annotated and classified for the final purpose of creating a datasets from these annotations.
The main characteristics needed for this web application are the possibility to:

- Create campaigns, based on geographic locations
- Let users cooperate in the classification and annotation tasks
- Create annotations, drawing polygons on images and associating a label for each of them
- Supervise the annotations creation process
- Analyse statistics about the produced annotations

The tool we are proposing here can work in two different operating modes related to the tasks of classification and annotation of the images, or either both in a multi-stage process, that could potentially contain an illegal landfill.

The first mode is related to the tasks of "Selection" and after that "Annotation": the first stage acts as a classifying stage where it is possible to tell if the site represented in the image contains objects of interest or not, then in the second stage it is possible to annotate only the images that were positively flagged in the first stage.

The second mode is related to the task of "Annotation Only", in this case every image can be directly annotated, without a screening stage, and in case of an image that does not contain any element of interest it is possible to reject it without annotating it. Clearly rejecting an image acts like the classification stage of the "Selection" task where sites are first classified.

In any type of task the tool always shows one image, representing a site, at a time. The user, after an inspection and evaluation of the image, performs different actions depending on the tasks is completing. In case of a "Selection" task the user answers specific questions and then the tool saves the information related to those answers. In case of a "Annotation" task the user is asked to recognize inside the objects of interest drawing polygons to outline them.

4.3.1 Target groups

The application supports two types of users (as shown in Table 4.1): admin and worker. The purpose is to differentiate usage modes that are available to different kind of users and also to protect sensible data (e.g. illegal landfills sites coordinates or their statistics) as much as possible, sharing them only when necessary and to whom is authorized. Only worker users can Sign Up to the annotation tool directly online, admin user necessitates to be manually inserted by the administrators of the web application. This was done to prevent anyone from signing up as admin and creating their own campaigns, since the scope of the annotation tool, in this stage of the project, is to gather workers to annotate images for the creation of the dataset of interest for this project.

Role	Tasks
Admin	• administrates the creation of campaigns and the statistics available for every created campaign • decides which user can access which campaign • inspects and edits the annotations done by workers • exports annotations data for deep learning tasks
Worker	• accesses the campaigns upon admin authorization • classifies or annotates images • checks his personal statistics on tasks and modifies the results he produced

Table 4.1: Target groups description

4.3.2 Requirements

In this subsection are analysed the requirements of the web annotation tool, divided in categories depending on the user type associated:

- **generic user** (can be both admin or worker), that leads to the definition of Users Requirements (UR)
- **admin** user, that leads to the definition of Admin Requirements (AR)
- **worker** user, that leads to the definition of Worker Requirements (WR)

User Requirement UR#1: Sign In

As a user, both admin and worker, I can sign in to the annotation tool inserting my personal login information in the Sign In page.

User Requirement UR#2: Edit profile information

As a user, both admin and worker, I can edit my personal information contained in my profile. The editable field are full name and password.

Admin Requirement AR#1: Campaigns creation

As an admin I should be able to create campaigns, letting the users collaborate in annotating the site images contained in it with the final goal to create a dataset from this annotations.
A campaign is composed of a:

- **name**: to reference it,
- geographic **coordinates list**: used to create the images,
- type of **operating task**: Annotation only or Selection+Annotation,
- list of **labels**: the name of the elements of interest to annotate,

- **source** of images: Mapbox or Orthophoto,
- **distance**: from the center location used to generate the images,
- **threshold** of needed annotations: images with less than this number of annotations are not considered in final statistics
- **state**: ready, started, ended

The state of the campaign is important because it determines what actions are possible to execute:

- **ready**: the campaign can be modified and users are not able yet to contribute. This is the default state when the campaign is just created.
- **started**: there are certain aspects of the campaign that are not longer editable and users can start collaborating. The owner of the campaign has to manually start the campaign.
- **ended**: the campaign is no longer editable and users can no longer collaborate to it, but the admin can still edit annotations. This state is reached when the owner manually closes the campaign.

Admin Requirement AR#2: Campaigns edition

As the admin owner of a campaign I should be able to edit campaigns, to adjust its fields because of previous errors or missing considerations. The name field of the campaign can be edited in any moment independently of the status of that campaign, instead labels can be edited adding and removing them when the campaign is in "ready" state otherwise if the campaign is in "started" state only new labels can be added.

Admin Requirement AR#3: Change campaign status

As an admin I can start and close campaigns.

Admin Requirement AR#4: Grant to workers access to campaigns

As an admin I can grant to worker access to the campaigns. The reason behind this functionality is to allow the admin to keep sensitive locations private and decide which worker can access specific campaigns. In our scenario a possible example is represented by the sites provided by ARPA.

Admin Requirement AR#5: Inspect available sites on map

As an admin of a campaign I can inspect all the sites contained in a specific campaign through an interactive map and see the number of annotations done for that site. This functionality is important because allows the admin to effectively inspect the territorial distribution of the site that belongs to a campaign.

Admin Requirement AR#6: Inspect campaign statistics

As an admin I can view all the statistics of a single campaign. The statistics available are:

- campaign progress
- accepted/rejected site percentage:
- total number of image annotated done (this considers all the users images annotated)
- list of labels with their total count
- user personal stats on the campaign

The importance of this functionality is given by the fact that an admin has the need to know the progress of a campaign in term to images annotated and labels used throughout the process done by the workers.

Admin Requirement AR#7: Inspect worker annotations

As an admin I can inspect the annotation results (all the annotations done for a specific site image) done by the workers for every site contained in a campaign I own. More than this it is possible to see the statistics about that site:

- accepted and rejected count
- distinct count of the labels

This functionality is very important since is the one that allows the admin to directly check the annotations done by the workers.

Admin Requirement AR#8: Edit any annotation from workers

As an admin I can always access to all annotation results and also edit them if they contain annotations that are not well done and that can introduce noise in the dataset generated. The motivation behind this feature is to allow the **admin** to correct annotations since he **is the one concerned about the integrity of the dataset**, so it is up to him to decide, based on the specific scenario, the actual needs of the dataset and modify annotations to fulfill them.

Admin Requirement AR#9: Export results

As an admin I can export annotation results done by worker to use them. Annotation results can be exported in different ways:

- **a single result** when inspecting an image
- **all the results** related to a **specific campaign**
- **all the results** aggregated from **all the available campaigns**

The functionality of exporting one single annotation is intended for fast checking that annotation on another external geojson viewer
(e.g. www.geojson.io developed from Mapbox).
The other export functionalities are intended to export data that are going to be used for dataset creation purposes.

Worker Requirement WR#1: Sign up as a worker

As a user, with the intention to participate to the project, I can sign up as a worker in the web annotation tool to be able to perform classification and annotation tasks on the campaigns that will be available.

Worker Requirement WR#2: Classify sites

As a worker I can classify site images, in selection tasks, answering specific questions to tell if an element of interest is contained inside of an image or not and to tell if the image is corrupted. This simple classification allows, in a later step of annotation, to only annotate sites that are already tested to contain an object of interest and whose image is not corrupted. Some images that come from Mapbox provider can be corrupted or deteriorated by the preprocessing done by the provider itself, in those cases is preferable to not choose those images, thus explaining why there is a way to discard them.

Worker Requirement WR#3: Annotate sites

As a worker I can annotate site images detecting and outlining elements of interest that can be found inside the sites. This functionality is actually one of the most important since is the one responsible for the actual creation of the annotations that will compose the dataset.

Worker Requirement WR#4: Inspect his own annotations with their statistics

As a worker, for each campaign, I can inspect my own annotations also viewing: the count of labels used for each image, how many images I have already annotated and how many images are still available to annotate.

Worker Requirement WR#5: Edit his already annotated images

As a worker, for each campaign, I can edit my own annotations. This feature is important for example because as the time passes doing annotations the worker increases his expertise in recognizing interesting elements therefore he can go back to previously done annotations and improve them. Another example is the introduction of a new label, in that case the worker can go back and edit already done annotations using the new label. This functionality allows to avoid the creation of a whole new campaign with the same sites and the new label added, thus allowing to easier improve the quality of the annotations.

Worker Requirement WR#6: Export annotation results of his own annotation

As a worker, for each campaign, I can export, one by one, my own annotations from any site image I have annotated in geojson format. This functionality allows the worker to view the annotations he has done in an external geosjon viewer (e.g. www.geojson.io developed from Mapbox).

4.4 Tagger tool: Technologies

4.4.1 Node.js

Node.js[6] is an open-source, cross-platform, JavaScript runtime environment that allows to execute JavaScript code outside the browser, thus allowing it to be used to also develop server applications. Being able to run server-side Node.js represents the "JavaScript everywhere" paradigm, that unifies the web-application development around a single programming language.

Node.js runs on the V8 JavaScript runtime engine, that takes the JavaScript code and converts it into a faster machine code. Once converted in machine code since it is low-level, the computer can run it without needing to first interpret it.

What makes Node.js really powerful and appealing is the fact it uses an event-driven, asynchronous model that makes its use lightweight and efficient; more than this its package ecosystem, npm, is a great ecosystem of open source libraries, that enables to expand it with any kind of needed functionality.

Node.js operates on a single-thread event loop, using non-blocking I/O calls, allowing it to support tens of thousands of concurrent connections without incurring the cost of thread context switching. The design of sharing a single thread among all the requests that use the observer pattern is intended for building highly concurrent applications, where any function performing I/O must use a callback function to define what to do when the asynchronous function has been completed.

The event loop is a design pattern, that waits for and dispatches events or messages in a program, allowing great scalability, not needing to use more processes or threads to accomplish the same results.

Node.js allows also the adaptation of server-side development patterns such as Model-View-Controller (MVC), Model-View-ViewModel (MVVM).

[6] https://nodejs.org

The main benefits of Node.js are summed up in this list:

- the ease of coding in just one programming language both client and server side,
- the high performance that an asynchronous execution allows,
- the possibility to implement the front end using a single-page application paradigm corroborated by Model-View-ViewModel pattern,
- the great availability of already published libraries that allow to implement any needed functionality

Node.js has been chosen as main technology in this project such as runtime environment.

4.4.2 Single Page Application (SPA)

A single-page application is a web application that interacts with the user by dynamically rewriting the current page rather than loading entire new pages from a server.

Typically in a single-page application, all necessary code (JavaScript, HTML and CSS) is retrieved when the page is loaded the first time, while appropriate resources are dynamically loaded and added to the page in response to user actions.

The page does not reload at any point in the process. For this reason single-page application is fast, as most resources are loaded typically once throughout the lifespan of the application(to the only downside that initial load is slower since a lot of resources need to be loaded at once), data are fetched in background, and individual user actions are more responsive since full page reloads are rare.

Interaction with the single-page application often involves dynamic communication with a web server behind that act as back end server. Only data is transmitted back and forth.

The development is simplified and streamlined. There is no need to write code to render pages on the back end server, which makes the development much easier.
Single-page application supports:

- rich client-side functionalities that does not require reloading the page as users take actions or navigate between areas of the app
- rich client-side behaviors much more ready than traditional applications

Often single-page applications are used alongside a particular architectural and software pattern called Model-View-ViewModel (MVVM) that using ViewModel is best suited for this kind of applications.

The reason why has been decided to develop the front end of the web application as a SPA is because it allows creating rich client-side functionalities very reactive to users inputs, both things useful for the purpose of this project that incorporates interactive maps and drawing tools. Another reason was to simplify early development stages not requiring necessarily an already developed back end server. This decision allowed also to decouple back end server and code components needed to render the page from each other allowing to move them only in the front end.

4.4.3 Model-View-ViewModel (MVVM) pattern

Model-View-ViewModel (MVVM) pattern is a variant of the frequently used Model-View-Controller (MVC) pattern. The difference between this two patterns is the presence of the ViewModel instead of the classical Controller.
Model-View-ViewModel is composed of three major component:

- Model: represents the actual state containing the data of the application. The model only holds the information but neither behaviors nor services that manipulate the information.
- View: is the graphical representation of how the model data are shown to the users and it receives the user's interaction and forward it to the

right component that can handle it. In case of MVVM this component is represented by the ViewModel.

- ViewModel: acts as a binder between the Model and the View.

The ViewModel relies on the process of data binding that establishes a connection between the app UI and the data it displays, in this way the visual changes to the View reflects the changes in the Model. The way ViewModel works is similar to the one of Controller does it in MVC. So if they are so similar it is fair to ask why the MVVM variant came up. The answer lies in the fact the MVC design pattern still has drawbacks in some cases. For example it is standard that each model in a database has its own controller, so when an application scales and evolves with many related models, the amount of controllers used must grow in tandem. This, coupled with the natural introduction of new layers of abstraction brought on by most frameworks, creates a codebase that can becomes very difficult to navigate through.

ViewModel is an efficient way to overcome this issue, since it relies more on the front end rather than to the back end. This is also one of the major reason why it has been chosen as architectural pattern for this project.

4.4.4 Knockout

Knockout[7] is a JavaScript library that implements the MVVM pattern allowing the use of templates. This framework allows to easily implement also the single-page application paradigm, since such templates simplify the way the programmer modifies the pages. With Knockout it is possible to create rich, responsive user interfaces with a clean underlying data model. Any time that the UI changes Knockout dynamically updates and implements changes using data-binds.
Knockout offers many features and benefits:

- dependency tracking: automatically updates the right parts of the UI whenever the data model changes

[7] https://knockoutjs.com/

- declarative bindings: represents a simple and obvious way to connect parts of the UI to the data model
- extensible: it is possible to implement custom behaviors as new declarative bindings for easy reuse in just a few lines of code
- abstraction: works with any server or client-side technology and can be added on top of an existing web application without requiring major architectural changes
- lightness: compact and lightweight
- compatibility with mainstream browsers: IE 6+, Firefox 2+, Chrome, Safari, Edge

Knockout allows to implement the front end as a single-page application that uses MVVM pattern, has great compatibility with a lot of browsers (even their old versions) and is very lightweight. For all those reasons has been chosen as main front end library.

4.4.5 Leaflet

Leaflet[8] is an open-source JavaScript library for interactive maps, designed with simplicity, performance and usability in mind. It works efficiently across all major browsers, can be extended with lots of plugins, has an easy-to-use and well-documented API. It is very lightweight but still has all the mapping features needed to implement powerful and interactive maps.

In a web annotation tool that aims to create annotations upon geographical data the use of a library like Leaflet is fundamental. In this case scenario Leaflet is used both to implement geographical maps that use tile layers and also image containers for custom image layers created from single images.

When using a custom image as image layer Leaflet allows the user to zoom in and out the image. It is possible to drag the zoomed image thanks to a dragging customizable feature that is also able to recognize when the

[8] https://leafletjs.com

user is going out from the image boundaries. When this happens the image is again repositioned at its borders.

More than this Leaflet can also be powered and extended with multiple plugins. In this specific project the plugin Leafet Draw was added.

Leaflet Draw allows to easily extend the functionalities of the map enabling the user to draw polygons and markers on the map. It also allows extracting in an object everything has been drawn which is extremely useful for the proposed web application of this project.

4.5 Tagger tool: Front end

The Front end the of web application has been developed as a single-page application with the Model-View-ViewModel pattern using Knockout.js as major developing library, and it represents what the users interact with while working on the web annotation.

Since the front end is a SPA there is only one page that actually rewrites its contents depending on the inputs received from the users and the contents dynamically got as response from the back end server. Although to the final users, unaware of this technical feature, these rewriting actually feels like the loading of different pages. For this reason to simplify the reading of this section we will consider the concept of page referring to what the users see rather than what is actually happening at code level.

The pages of the web application are divided in two major distinct areas: one for admin and one for workers. The admin area is more focused on the concept of campaigns, since they are created only by admin users, while workers area is more focused on the concepts of tasks since the users will primarily work on the tasks of selection and annotation.

The only three exceptions to this sub-division are represented by the Sign Up, Sign In and Profile pages.

4.5.1 Common area

Sign Up page

This page is intended to fulfill the *Sign up as a worker* WR#1 requirement. In this page the users can enroll to the web annotation tool effectively becoming workers for this context. The fields necessary to register a new worker are: First name, Last name, Username, Password.

Username and password represent the login elements used by the worker to Sign In in the application for this reason the username field has to be unique to identify one specific worker among all the others. In case the user chooses an already took username a message is prompted below the field notifying it and telling him to choose a new username.

Password field since is crucial for Sign In has a secondary confirmation field that requires the user to double insert and check his password.

The first and last name fields when the Sign Up is successfully completed are merged up in the full name field that will be later shown in the Profile page.

Sign In page

This page is intended to fulfill the *Sign In* UR#1 requirement. This represents the landing page that is always shown when a not signed in user opens the website and where the users can sign in to the annotation tool inserting their username and password fulfilling

Profile page

This page is intended to fulfill the *Edit profile information* UR#2 requirement. In this page the user can edit its full name and password, the latter one requires to be confirmed and written two times as in the Sign Up page.

4.5.2 Admin area

The area created for admin user is composed of several pages voted to campaign creation, statistics analysis and results export related to the currently created campaigns. First an admin cannot directly sign up on the website, he needs to be manually inserted as admin users by the administrator of the website. This is because the objective is not to let everyone be an admin and create campaigns, at the moment this tool is meant more for gathering worker users rather than admin users that will create campaigns for their scenarios. Of course this sign up limitation is functional to our scope, but not necessarily something that won't be changed in the future with new requirements available. Once the admin log in he found himself in the home page.

Admin home page

This page is intended to show the list of campaigns and for each of them the possible actions that can be performed:

- start campaign creation process
- inspect campaign information
- inspect active/inactive workers list
- inspect campaign site location on interactive map
- inspect campaign statistics
- export worker's results

Each action, except export results ones, brings the admin to a different page. In this page results can be exported by campaign clicking the export button of the relative campaign, otherwise it is possible to export all the results of all campaigns at once. This page fulfills *Export results* AR#9 requirement.

Campaign creation page

This page is intended to fulfill the *Campaign creation* AR#1 requirement. For this purpose, the page contains a form with all the information needed to define a campaign. Such form contains input text/number for the basic attributes (e.g. name, labels, threshold) and radio buttons for the fixed attributes (e.g, distance, tasks, images sources).

As mentioned in the AR#1 the main component of the campaign is a list of sites. To upload such list, this page contains a field to upload a comma separated file (csv).
An example of valid file would be:

```
Id, Latitude, Longitude
0, 45.9, 9.6
1, 45.8, 9.5
```

When the button to create the campaign is clicked, the correctness of the fields is checked (a more robust validation about the dataset is done in the back end server), and the data is sent to the server to create the campaign. If successful, the user is redirected to his Home page, otherwise proper messages are shown.

To select the distance there is a radio button with the multiple choice, the distance is expressed as the edge of the image, e.g. 0.5km would mean that the image represents an area of 0.5km per edge and thus 0.25km^2. In this case scenario it has been decided to work with two fixed distances: 0.5km and 1km. In future implementations of this web application it will be possible to change it adding more choices, or letting the admin choose it directly for example.

The labels represents the classes that are interesting to find inside the images that will be made available during the annotation task inside a popup, that is created when a polygon or a marker is drawn on map.

Campaign info page

This page is intended to fulfill the *Campaigns edition* AR#2 and *Change campaign status* AR#3 requirements. Inside this page there is a form in which are contained all the information of the specific campaign. In this page it is possible to:

- Edit campaign name and labels list
- Start "Ready" campaigns
- End "Started" campaigns

From this page it is possible to directly reach the *Campaign images and stats page* and the *Workers management page* of that campaign.

Campaign workers management page

This page is intended to fulfill the *Grant to workers access to campaigns* AR#4 requirement. In this page for workers management there a list of active/inactive workers with the related tasks for which they are enabled/disabled to. For example in case of "Selection+Annotation" tasks an user could be enabled to only do selection or even only annotation. Worker can be added anytime, even if the campaign is already started, but can be removed only if the campaign is not yet started.

Campaign images and stats page

This page is intended to fulfill the *Inspect available sites on map* AR#5 and *Inspect campaign statistics* AR#6 requirements.

In this page there is an interactive map where the admin can inspect site locations showed as markers. When a marker is clicked the map is zoomed in to that location and a squared area around the site and a popup with the site stats are shown. The squared area represents how much distance is shown to the worker, through the image, in selection/annotation task.

From the popup it is possible to reach Admin Image detail page for that specific site.

Always from this page the admin can see the campaign statistics that comprehends.

Admin image details page

This page is intended to fulfill the *Inspect worker annotations* AR#7 and *Edit any annotation from workers* AR#8 requirements.

In this page is shown the image of the site location with its annotations and the statistics related to that site:

- how many times has been annotated, accepted and rejected
- the count of each label contained in the current annotation result shown in the image

Inside this page the admin can:

- **scroll among sites and annotations**: the admin can comfortably scroll between images, representing the sites, first and then once chosen the interested site image can scroll between its annotation results done by workers.

- **search by ID**: an alternative way of search that allows the admin to search sites by image ID.
- **results editing**: annotation results done by any worker can be edited by the admin that in order to do this he needs to modify the annotation result that needs a modification and save it; in future versions we could introduce this edition setting as optional at campaign creation time.
- **result import**: annotations can also be imported from a geojson, this geojson needs to be with WGS84 coordinates or pixel coordinates.
- **result export**: it is possible to export the currently selected annotation result in a standard geojson in WGS84 coordinates to be viewed in an external geojson viewer if needed.
- **inspect the site in Google Map**: pressing the Gmaps button opens a new tab in the browser to inspect the site location directly in Google Maps.

4.5.3 Workers area

The area created for the worker users is composed of several pages that are focused on allowing workers to perform the tasks that are enabled to them. Workers, differently from admin user, can directly sign up on the web application without having to be manually inserted by website administrators. Once a worker has signed in will be redirected to his home page.

Worker home page

This page is intended to list all the available campaigns where for each of them the worker can:

- Start the task session
- Inspect annotations done with their stats along with possibility to edit them

Selection page

This page is intended to fulfill the *Classify sites* WR#2 requirement. In the page for "Selection" tasks a Leaflet container is loaded with inside the image that the worker need to inspect and upon that image there is a question to answer with a multiple choice mode; there can be more than one question in a row that are sequentially shown as soon as the worker advances answering the previous ones. The worker can in any moment navigate to previously answered question and change his answers before submitting the results related to that image.

Annotation page

This page is intended to fulfill the *Annotate sites* WR#3 requirement. In the page for "Annotation" tasks a Leaflet container, with the drawing plugin too, is loaded with inside the image that the worker needs to annotate. The drawing plugin allows the worker to draw polygons and markers in the image.

Once a polygon is drawn a popup is rendered on that polygon with a drop-down list of the labels that are available for that campaign.

Two buttons "Send" and "Reject" are available respectively to send the annotations done to the server saving them and reject an image that does not contain anything of interest.

There two directional buttons to iterate through the site images to temporarily skip them. This can be useful in case the worker is stuck doing an image hard to analyse.

In this page it is possible, by pressing the Gmaps button, to open a new tab in the browser to inspect the site location directly in Google Maps. The point of this functionality is to provide a further source to the worker to check the area in an external interactive map that has also support to multi-temporal data, in order to help him in identifying elements that are hard to recognize.

Worker image details page

This page is intended to fulfill the *Inspect his own annotations with their statistics* WR#4, *Edit his already annotated images* WR#5 and *Export annotation results of his own annotation* WR#6 requirements.

In the page a Leaflet container, with the drawing plugin too, is loaded allowing the worker to iterate through the site images, associated with the campaign he selected. For each site image it is possible to view all the annotations done with the count of the labels used in that annotations and personal statistics of the worker for that campaign: how many images annotated and how many to annotate yet.

In this page it is possible, by pressing the Gmaps button, to open a new tab in the browser to inspect the site location directly in Google Maps.

Always in this page, thanks to the drawing plugin the worker has the possibility to edit his own annotations. With the available editing options the worker can:

- delete already existing elements in the annotation result
- modify the shape of an already existing annotation

- draw new elements in the annotation result
- change labels associated with existing drawn annotations

In the case scenario of the "Selection+Annotation" task if the image sites have been only selected and not annotated yet only the stats relative to the personal progress of the worker in that campaign are shown.

Otherwise if at least one site image has been annotated the rest of the page with image container with the drawing plugin and labels counter is loaded, allowing the worker to perform the actions just described.

4.6 Tagger tool: Back end

The Back end of the web application, that deals with all the data recovering and persistence, is represented by a REST Service developed with Node.js as a runtime library.

4.6.1 Structure

The back end server has various API call available divided in the different classes:

- **auth**: for authenticating and allowing requests of other APIs
- **campaign**: for all the requests related to campaigns, only admin has access to this API
- **task**: for all the requests related to tasks, only workers have access to this API
- **user**: for all the requests related to users
- **image**: for the requests related with the retrieval of satellite imagery from Mapbox Static Images API (described in subsection 3.1.2)
- **orthophoto**: for the requests related with the retrieval of aerial orthophoto imagery from the Orthophoto service (described in section 4.2)
- **results export**: for the requests related with the export of annotation results done by workers in the campaigns, only admin has access to this API

Auth

This API is specifically created to authenticate users requests in the website and also to allow the functions of login and logout.

Campaign

This API is specifically created to let admin operate on campaigns allowing:

- creation
- editing
- status management (start and termination)

During the campaign creation phase, pre-processing functions to check the presence of all the necessary fields and the correctness of data inside these fields are invoked.

Some check examples are the correctness of the csv format for the list of the locations or the coordinates validation process that verifies that latitude and longitude coordinates are in their validity range of degree.

Also everything directly connected with campaigns can be found in this API, like functions to:

- enable/disable workers for a specific campaign
- get the list of images of a specific campaign
- get statistics of a single image including associated results
- edit workers results as admin

All these functions check if the operation is allowed and is done only to the elements referred to a specific admin. This means that an admin cannot act on campaigns of other admin users.

When a result is edited, by an admin, the contents of the result, in this case geojsons, are validated and a new field, containing the time information of the last update specific for the admin, is set and then the result is updated on the database. The back end controls for the editing of annotation results, as admin, follow these steps:

- check if the type of the user is admin
- check if the admin is the owner of the campaign to which the site image of the annotation result belongs
- check if the annotation result is a valid and well formed geojson

Task

This API is specifically created to let the workers operate on tasks they are assigned on. It mainly allows to:

- get the images to classify/annotate, chose from the scheduler
- get the statistics of a specific task
- edit workers results

When the worker starts a classification or annotation task for the first time or when a result is sent to be saved the session related to that task, and that specific user, is updated and a scheduler function choose a new image to be selected as next image on that task.

The scheduler works taking into account also the other workers that are enabled on that same task. In this way it is possible to select a new image that has not been processed yet.

This allows to process all images once before an already processed one is selected as next image, enhancing the collaborative feature of the web annotation tool and remaining transparent to the workers.

The whole point of this feature is to speed up the process of classifying/annotating images to complete them all with at least one annotation result as soon as possible, in order to have ready results faster, without avoiding the possibility of having all the workers completing all the available images.

Upon receiving the results from classification/annotation tasks there are controls that make a validation before persisting data in the database. The back end controls for the saving of classification/annotation results follow these steps:

- check if the worker is enabled for that campaign
- parse of the result to verify if is valid to be persisted in the DB

When a result is edited, by a worker, the contents of the result are validated and a new field, containing the time information of the last update specific for workers, is set and then the result is updated on the database. The back end controls for the editing of annotation results follow these steps:

- check if the worker is enabled for that campaign
- check if the annotation result is a valid and well formed geojson

User

This API is specifically created to manage the users of the web application. What this API allows is to:

- sign up workers
- get users information
- let users edit their profile information

Image and Orthophoto

Those two APIs are specifically meant to deliver an image upon a called url containing the ID, as saved in the database, of the image that is needed. One API is developed for Mapbox map provider (Image) and the other one is developed to provide images that comes from the Orthophoto service.

The Image API, since it uses Mapbox, internally retrieves the coordinates (related to the requested image from the database) and other information, like distance and Mapbox API access token (required to authorize the requests) to construct the query to the external Mapbox Static Images API, described in sub-section 3.1.2, that will produce the image.

The Orthophoto API instead call the Orthophoto service, described in section 4.2, that retrieves the right image depending on coordinates and distance parameters.

Both these APIs retrieve an image that will be then sent back to the front end and loaded by Leaflet image container as an ImageLayer object.

Result Export

This API is specifically created to let the admin export results done by the workers that are using the web application. It is possible to export the results of a single campaign or even the entire collections of results.

These features have been developed to make easier for the admin to use the data developed by workers.

Exported data contain the annotation results with all the required information about their related site images.

When exporting the results of a single campaign it is performed a check which verifies that the admin that is authorizing the request is the owner of that specific campaign.

When exporting all the available results from all the available campaigns, it is performed a check that only retrieves the campaigns of which the admin, that is currently authorizing the request, is the owner.

Chapter 5

Achieved Results

This chapter covers a typical workflow realized utilizing the web annotation tool presented in Chapter 4 and also a qualitative and quantitative analysis of the annotation results obtained during the whole annotation process that brought to the creation of the final dataset.

5.1 Tagger tool

In this section we guide the reader through the tool describing a typical workflow, from the creation of a campaign, to the annotation process and the inspection of the annotation results. The workflow will be explained through the description of the main use cases of the annotation tool shown along with the images of the pages used in the process. A demo video tour is also available online directly in the web annotation tool, without requiring any registration, at `http://tagger.como.polimi.it/#!/demo`.

Campaign creation

This scenario covers the Requirement AR#1.

Users	Admin
Precondition	The user is logged in with admin role.
Workflow	1) The user clicks the "Create new campaign" button in the *Admin home* page (Figure 5.1). 2) The *Campaign creation* page (Figure 5.2) is shown. 3) The user inserts all the data needed in the apposite form and then clicks the "Create" button. 4) Data is validated and a request is sent to the server.
Post-condition	The campaign is persisted in the database and user is redirected to the *Admin home* page. Here the new campaign is listed with "Ready" status.

Table 5.1: Use case: Campaign creation

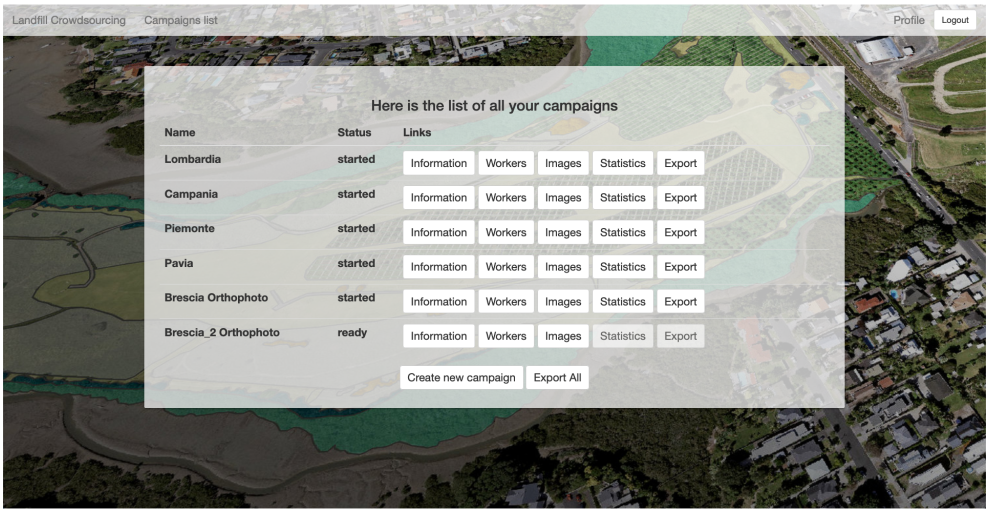

Figure 5.1: Admin home page

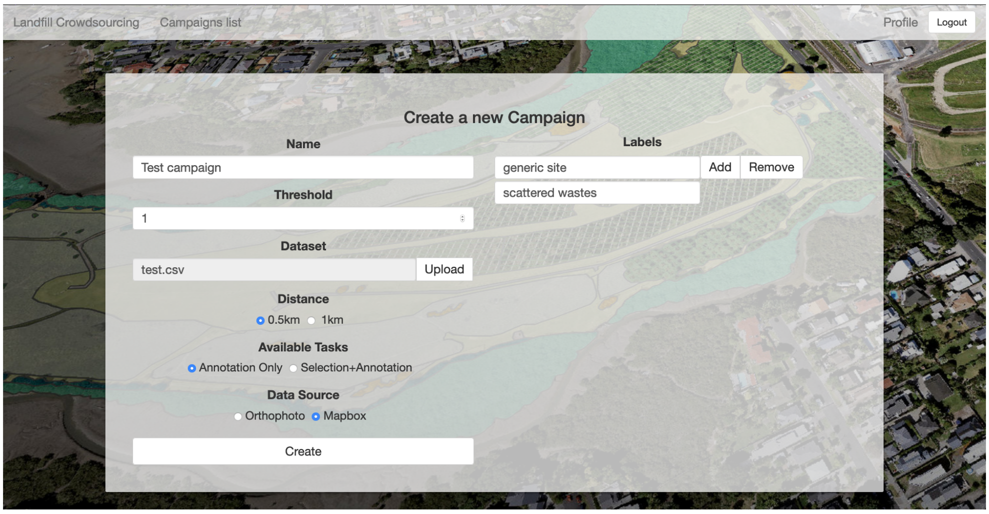

Figure 5.2: Campaign creation page

Enable users in the campaign

This scenario covers the Requirement AR#4.

Users	Admin
Precondition	The user is logged in with admin role. The campaign is created.
Workflow	1) The user clicks the "Workers" button referred to the campaign of interest in the *Admin home* page (Figure 5.1). 2) The list of workers for the selected campaign is shown in the dedicated *Campaign workers management* page (Figure 5.3). 3) The user clicks the "Enable/Disable" button of the desired worker.
Post-condition	The collaboration status of the selected worker has changed (enabled/disabled) for the given campaign.

Table 5.2: Use case: Enable users in the campaign

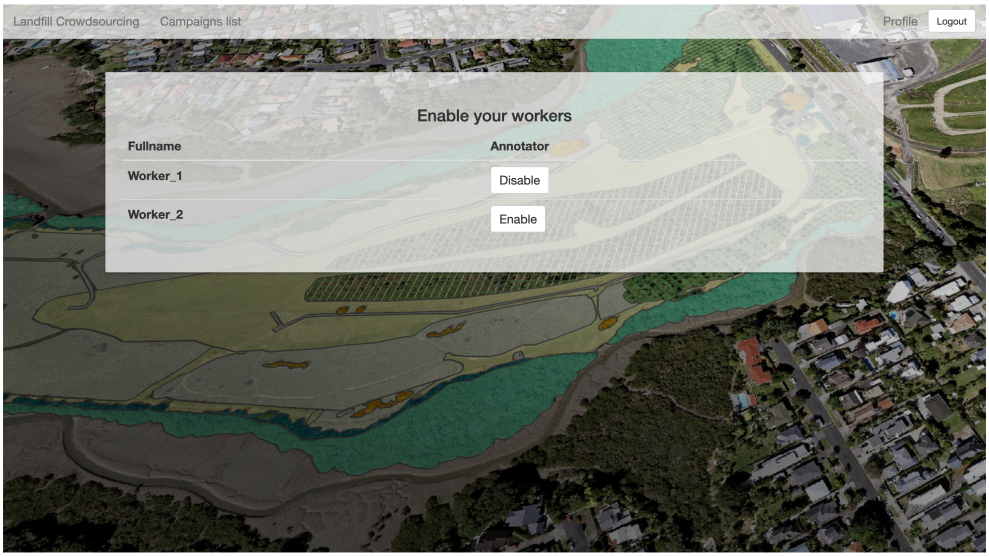

Figure 5.3: Campaign workers management page

Start campaign

This scenario covers the Requirement AR#3.

Users	Admin
Precondition	The user is logged in with admin role. The campaign is in "Ready" state.
Workflow	1) The user clicks the "Information" button of the campaign of interest in the *Admin home* page (Figure 5.1). 2) The *Campaign info* page (Figure 5.4) is shown. 3) The user clicks the "Start" button to start the campaign.
Post-condition	The campaign is started and the change is persisted in the database. The status of the campaign changes from "Ready" to "Started". Users can start collaborating.

Table 5.3: Use case: Start campaign

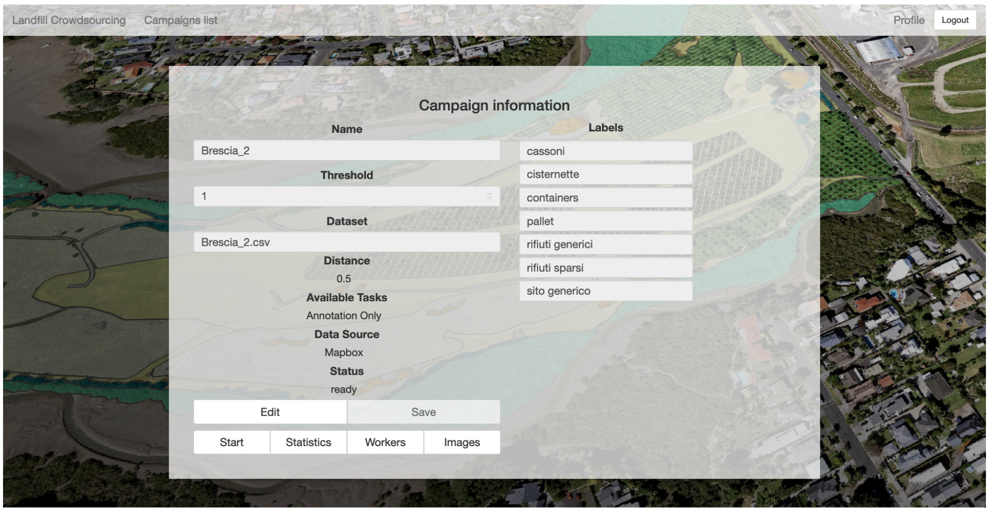

Figure 5.4: Campaign info page - ready state

Annotate an image in a campaign

This scenario covers the Requirement WR#3.

Users	Worker
Pre-condition	The user is logged in with worker role. The campaign is in "Started" state. The user is enabled to participate in the given campaign.
Workflow	1) The user clicks the "Start session" button referred to the campaign of interest in the *Worker home* page (Figure 5.5). 2) The *Annotation* page (Figure 5.6) is shown and the image container, with inside the image and the drawing plugin, is loaded. 3) The user draws a polygon for an identified element of interest. 4) The user selects a label for the created polygon from the popup that appears upon it. 5) Repeat steps 3) and 4) until all the identifiable elements are annotated. 6) The users clicks the "Send" button to save the annotation of all elements present in the image.
Post-condition	The annotation is persisted in the database. The user is presented with a new image to annotate. In the case the worker completed the campaign, he is redirected to the *Worker home* page.

Table 5.4: Use case: Annotate an image in a campaign

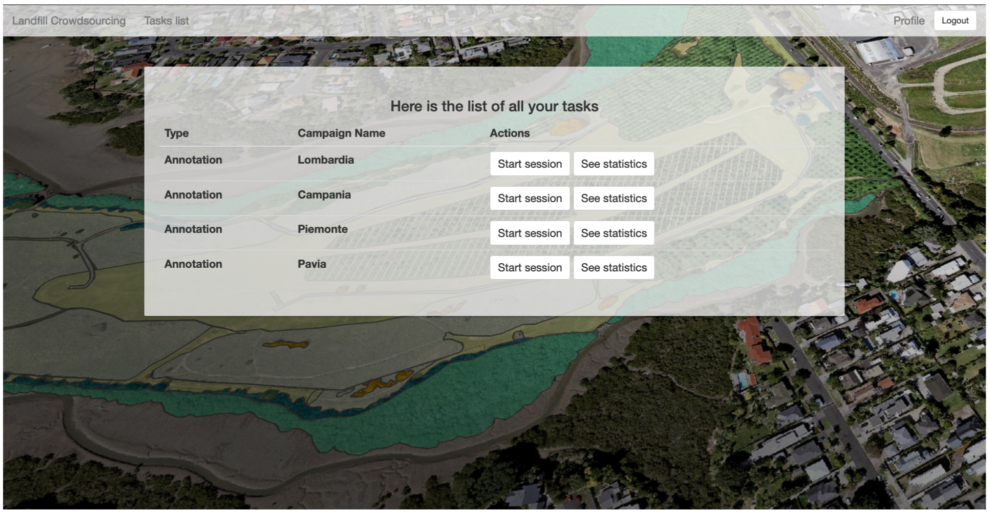

Figure 5.5: Worker home page

Figure 5.6: Annotation page

Reject an image in a campaign

This scenario covers the Requirement WR#3.

Users	Worker
Pre-condition	The user is logged in with worker role. The user is enabled to participate in the given campaign. The campaign is in started state.
Workflow	1) The user clicks the "Start session" button referred to the campaign of interest in the *Worker home* page (Figure 5.5). 2) The *Annotation* page (Figure 5.6) is shown and the image container, with inside the image and the drawing plugin, is loaded. 3) The user clicks the "Reject" button to the reject the image in which he cannot identify any elements of interest. 4) A popup of confirmation is shown. 5) The user clicks the "Yes, reject it" button to confirm his decision. **Alternate Flow:** 3) The user clicks the "Send" button without annotating any element. 4) A popup is shown alerting the user there are no annotations done, offering to reject the image directly from it. 5) The user clicks the "Reject" button to confirm his decision.
Post-condition	The annotation is persisted in the database. The user is presented with a new image to annotate. In the case the worker completed the campaign, he is redirected to the *Worker Home* page.

Table 5.5: Use case: Annotation of images in campaign

Inspect and edit an annotation

This scenario covers the Requirements WR#4 and WR#5.

Users	Worker
Precondition	The user is logged in with worker role. The campaign contains at least one annotation result of that user.
Workflow	1) The user clicks the "See statistics" button related to the campaign of interest in the *Worker home* page (Figure 5.5). 2) The *Worker image details* page (Figure 5.7) is shown. The image container, with inside the image and drawing plugin, is loaded and the statistics are shown. 3) The user navigates through the images (with the directional buttons). 4) The user modifies the annotations of an image: - adding/editing/deleting polygons - editing polygon labels 5) The user clicks the "Save" button to save the changes.
Post-condition	The new annotation result is persisted in the database overwriting the previous one. The statistics of the image are updated.

Table 5.6: Use case: Inspect and edit an annotation

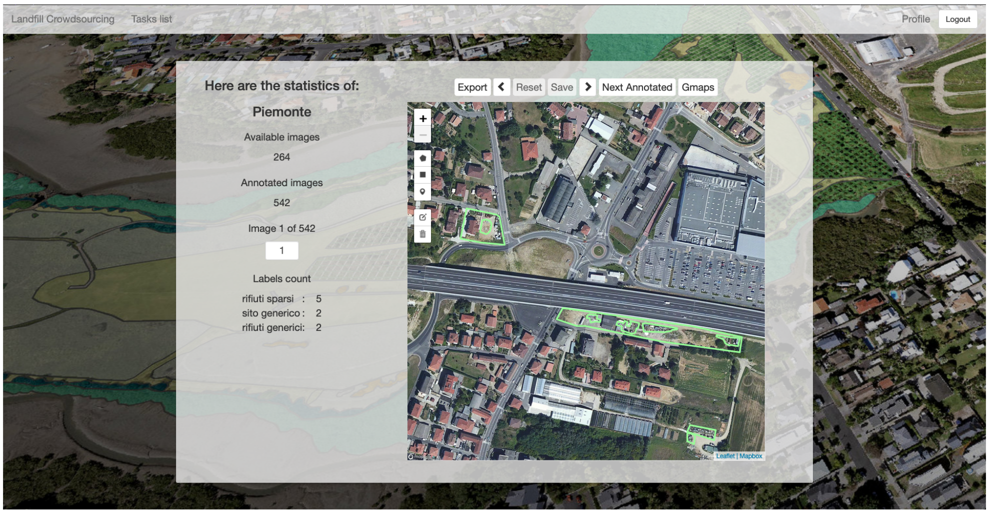

Figure 5.7: Worker image details page

Visualize statistics for the campaign

This scenario covers the Requirement AR#6.

Users	Admin
Precondition	The user is logged in with the admin role. The campaign is in status "Started" or "Ended".
Workflow	1) The user clicks the "Statistics" button referred to the campaign of interest in the *Worker home* page (Figure 5.5). 2) The *Campaign images and stats* page (Figure 5.8) is shown and the map with below the stats is shown, the page is automatically scrolled to the statistics. 3) The user inspects the statistics of that campaign. 4) The user clicks the "Update stats" button to update the statistics with the latest annotation result.
Post-condition	When the "Update stats" button is pressed the server computes the updated statistics by querying the database.

Table 5.7: Use case: Visualize statistics for the campaign

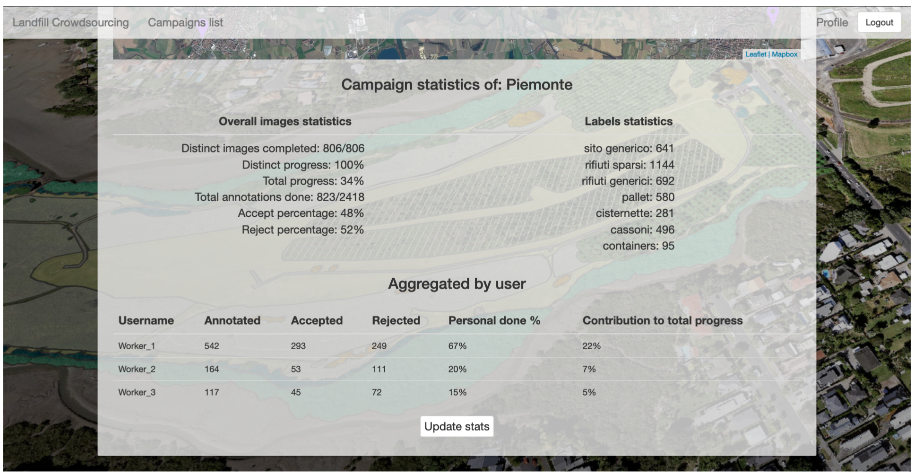

Figure 5.8: Campaign images and stats page

Inspect sites contained in a campaign

This scenario covers the Requirement AR#5.

Users	Admin
Precondition	The user is logged in with the admin role. The user is the owner of the campaign. The campaign is in status "Started" or "Ended".
Workflow	1) The user clicks the "Images" button referred to the campaign of interest in the *Admin home* page (Figure 5.1). 2) The *Campaign images and stats* page (Figure 5.9) is shown and the map is loaded with a marker for each site. 3) The user visualizes all the sites on map. 4) The user clicks a site marker.
Post-condition	The map is zoomed to the site of the clicked marker, the area of the image is shown around the site and a popup is opened showing how many annotation results that site contains.

Table 5.8: Use case: Inspect sites contained in a campaign

Inspect users annotation in a campaign

This scenario covers the Requirement AR#7.

Users	Admin
Precondition	The user is logged in with the admin role. The user is the owner of the campaign. The campaign is in status "Started" or "Ended". The user is already in the *Campaign images and stats* page (Figure 5.9).
Workflow	1) The user selects the site he wants to inspect by clicking its marker on the map. 2) The user clicks the "Show results" button in the popup. 3) The *Admin image details* (Figure 5.10) page is shown. The image container, with inside the image and drawing plugin, is loaded and the statistics are shown. 4) The user inspects site annotations with its specific stats. 5) The users scrolls between the annotation results of the site image shown. 6) The users scrolls between the images (with the directional buttons). 7) Repeat steps 4) 5) 6) until the user inspected all the desired annotation results.
Post-condition	

Table 5.9: Use case: Inspect users annotation in a campaign

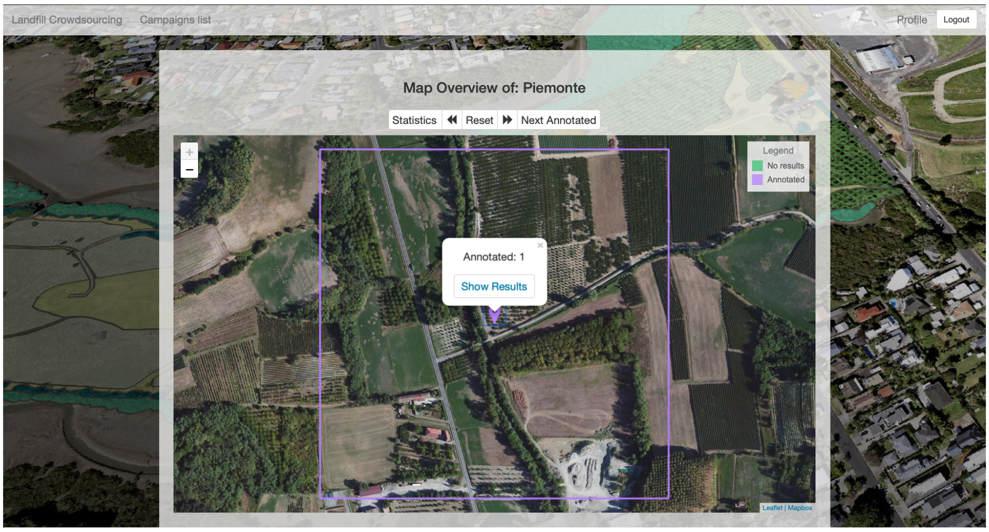

Figure 5.9: Campaign images and stats page

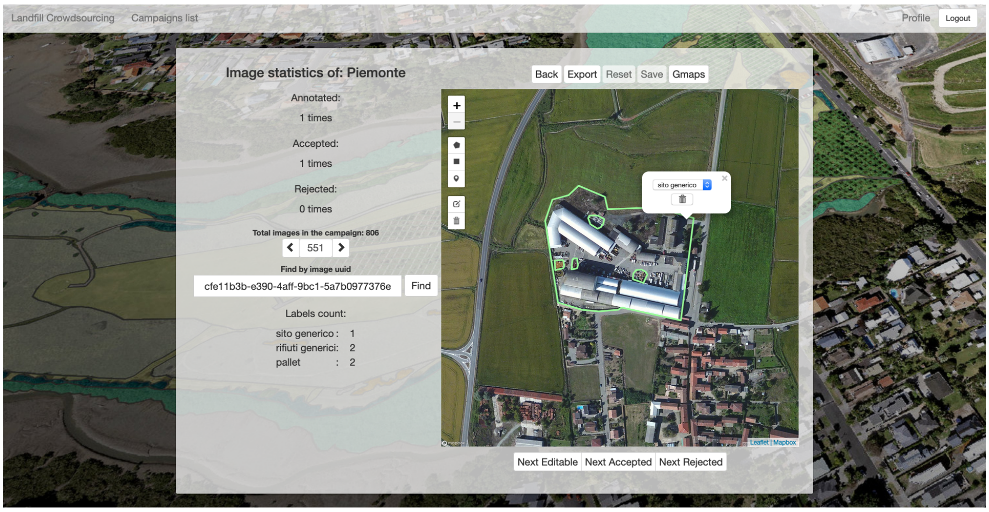

Figure 5.10: Admin image details page

Close campaign

This scenario covers the Requirement AR#3.

Users	Admin
Precondition	The user is logged in with admin role. The campaign is in "Started" state.
Workflow	1) The users clicks the "Information" button of the campaign of interest in the *Admin home* page (Figure 5.1). 2) The *Campaign Info* page (Figure 5.11) is shown. 3) The user clicks the "Terminate" button to terminate the campaign.
Post-condition	The status of the campaign now changes from "Started" to "Ended". The workers can no longer collaborate.

Table 5.10: Use case: Close campaign

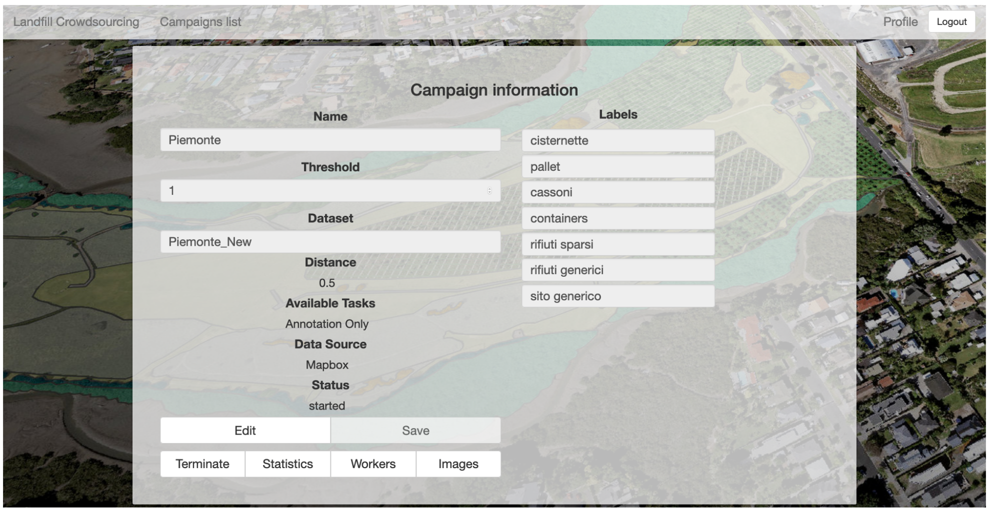

Figure 5.11: Campaign info page - started state

5.2 Illegal landfills dataset

This section presents a detailed analysis of the dataset introducing the statistics obtained about the annotations that compose it and a qualitative analysis of the image sources used for the realization of the dataset.

5.2.1 Dataset results

For the creation of this dataset we manually inspected 1529 images. Among these inspected images the 65%, 1000 images, were accepted since containing at least one of the classes considered in our research and the remaining 35%, 529 images, were rejected.

The 1000 accepted images were annotated with more than 10903 elements divided in 12 different classes.

In Figure 5.12 it is possible to see the count of the annotations divided per classes compared with the count of the images that contain an annotation of that class.

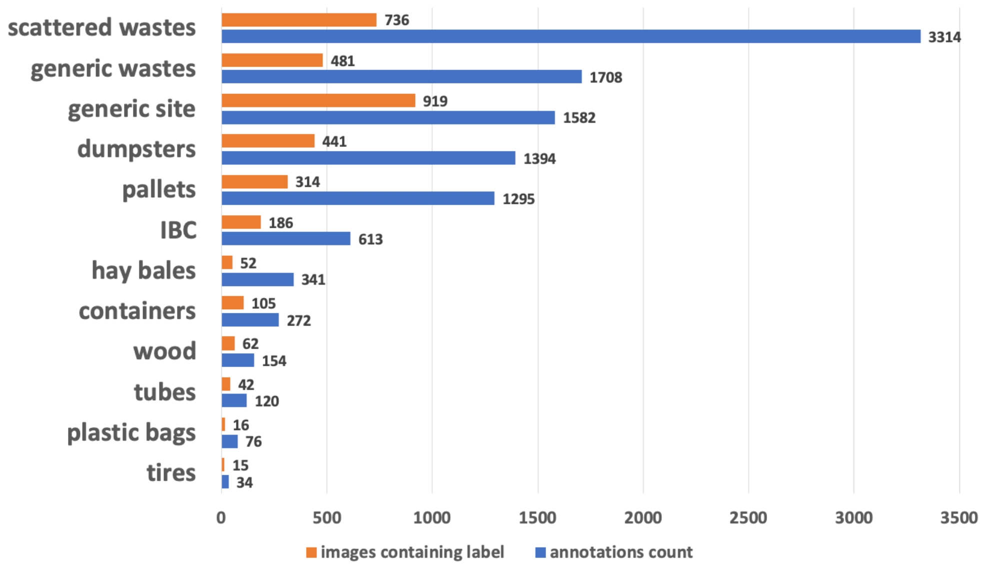

Figure 5.12: Count of annotations per class (blue) and count of the images containing an annotation with a specific class (orange).

The dataset is composed of polygons that delineate (segments) the objects of interest. In this section we will refer to such polygons as segmentation area. From such segmentation we could obtain a bounding box, useful for example, for object detection tasks in images. The bounding box (BB) is obtained by taking the most outer coordinate in each direction (top, botton, right, left).

Inside the images that we evaluated we found a total of 1582 generic site instances, representing a potential illegal landfills in a total of 919 images. Since the total number accepted images is 1000 this shows that in the 91,9% of the images annotated it was possible to identify an illegal landfill in addition to other wastes. An example of generic site annotated can be found in Figure 5.13.

Figure 5.13: Annotated example of generic site class

The class of wastes detected that has the highest count in the whole dataset, and that probably is the most representative of the illegal disposal issue, is the one of scattered wastes with 3314 instances found and annotated.

Among all the classes, except generic site, the class that appears the most in the images, with a count of 736, is the one of scattered wastes. An example of annotated scattered wastes can be found in Figure 5.14.

Other common found wastes in the annotated images are represented by: generic wastes, dumpsters, pallets.

In Figure 5.15 it is possible to see a couple of examples for each of the classes considered in the realization of this dataset.

Figure 5.14: Annotated example of scattered wastes class

Figure 5.15: Two annotated examples for each class of wastes.

In Figure 5.16 it possible to see the mean of the areas, in meters^2, of the annotations compared with the mean of the areas of the bounding box obtained from that annotations (both divided by classes), while in Table 5.11 it is also possible to see the minimum and maximum areas of the wastes always divided per classes and in meters^2.

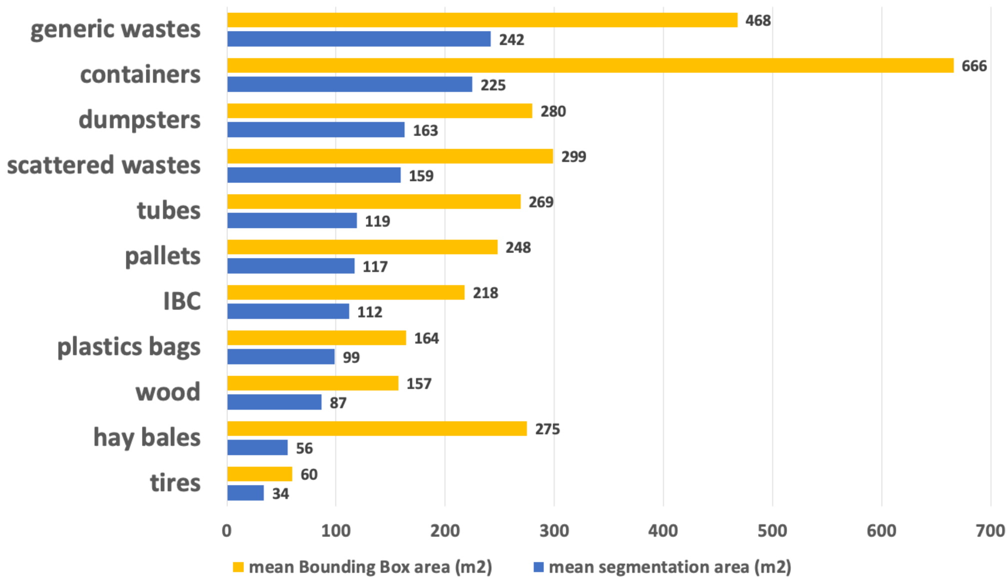

Figure 5.16: Wastes areas comparison of annotations per classes in m^2. The yellow bars represent the mean of the areas of the bounding boxes while the blue bars represent the mean of the areas of the segmentation of which the annotations are made.

As it was expected the areas of the bounding boxes, around the annotations, are greater than the one of the annotations for all the other classes. In Table 5.11 are also shown the areas information about the generic site class, from these data it is evident that the generic sites are on average always larger than the other classes of wastes.

Class name	min Seg area	max Seg area	mean Seg area	mean BB area
generic site	13	149939	11369	18672
generic wastes	3	42026	242	468
containers	4	6639	225	666
dumpsters	2	126048	163	280
scattered wastes	1	7791	159	299
tubes	3	1366	119	269
pallets	1	5615	117	248
IBC	1	2316	112	218
plastics bags	2	1708	99	164
wood	8	910	87	157
hay bales	2	495	56	275
tires	1	242	34	60

Table 5.11: Wastes areas comparison data per classes in m^2, with also minimum and maximum segmentation areas values.

5.2.2 Wastes characteristics analysis

The composition of the dataset, as in Figure 5.12, shows that there is an important presence of scattered wastes in the annotation results obtained from the inspected sites.

This can be explained by what scattered wastes actually represent: very small chunks of many different wastes, of different types and shapes, that when aggregated makes nearly impossible to recognize the single elements that compose them.

Scattered waste can be composed of: small domestic wastes, small junks and also small concentrations of rubble from old constructions.

Another interesting point that can provide an explanation to the strong presence of scattered wastes is given by the fact can be found both inside the illegal landfills and also outside of them, thus removing the potential assumption that scattered wastes can be found only inside illegal landfill. When found outside of landfills of course they are in small concentrations but can be potentially everywhere, as can be seen in Figure 5.17.

Figure 5.17: Scattered wastes outside a potential illegal landfill (left) and inside of an illegal landfill (right)

From these characteristics it is possible to understand that this typology of wastes is very easy to produce and moreover can be produced both from a single citizen, not caring about the environment, or even by criminal organizations that illegally stack them in hidden places, increasing very much the potential of find them in the territory.

The second most frequent class of wastes found is represented by the generic waste, that is our reference class for all the wastes that cannot be categorized by the other available classes, as mentioned in the section 3.2.1.

The strong presence of these wastes in the dataset can be explained by the fact that there are classes of wastes not considered during the annotation process that could be of interest in future, or by the fact we want to highlight that there is something suspicious on such location, but we are not able to identify the objects in particular.

As can be seen in Figure 5.18, representing an annotated example of generic wastes, it is not really easy to visually understand what are the constituting elements of this class. Is is only possible to tell that clearly are not scattered wastes.

This identification problem makes difficult adding new classes with the goal to reduce the number of wastes that ends in being classified as generic wastes. In future expansions of this dataset new classes will be added, after careful evaluation of which can reduce the usage of generic wastes thus improving the descriptive capabilities of the dataset.

Figure 5.18: Annotated example of generic wastes class

Another interesting result that came up from the counts of the annotations is the medium to high presence of: dumpsters, pallets and IBC. We found two reason as explanation for this result.

As anticipated in section 3.1.1, we tried to choose the lists of the suspicious sites with the goal to have the most uniform coverage of different types of territories: rural, urban and hybrid areas (a combination of rural and sub-urban areas). However the sites inside hybrid areas have shown the presence of wastes more typical for sub-urban areas rather than of a balanced combination of rural and sub-urban areas. It was not possible to evaluate this fact a priori without a full inspection of the sites in the lists.

The fact that wastes found inside hybrid areas are more typical of urban scenario, combined with the consideration that dumpsters, pallets and IBC are wastes more typical of an urban scenario, provides a first possible explanation to this medium to high presence of these classes.

The second explanation is that, unexpectedly, the assumption made about the more urban typical nature of these wastes is far less strict than we thought before starting the annotation process, thus allowing to find these wastes also in rural areas. A few examples of this atypical presence are provided in Figure 5.19 in which it is possible to see these urban typical wastes collocated in more rural scenario.

Figure 5.19: Example of atypical wastes in rural areas. In sub-image (a) it possible to see a group of dumpsters (in green) and a group of pallets (in red), in sub-image (b) there is a large group of dumpster (in red) found in an illegal landfill placed in a rural areas and in sub-image (c) it is possible to see two IBC (in red) with around scattered wastes in the middle of a wood. These annotations are opportunely made to provide insight only on these specific classes in rural areas and do not reflects annotation strategy used for the dataset.

5.2.3 Annotations color analysis

Since we annotated on images in the visible light spectrum and thus colored, we decided to study the dominant colors of the classes of wastes took in consideration for this dataset. To make this analysis we used a K-means clustering algorithm to extract, from the patches containing only the elements annotated, the 8 dominant colors for each class of wastes.
The results of this analysis can be seen in Figure 5.20.

From this analysis it is possible to see that the only two classes have dominant colors not in the black and white scale, namely: wood and hay bales. Their dominant colors are more on the brown scale and this is something we expected based on the observation we did during the annotation process. Moreover there is also another evident explanation given by the fact that these two classes represent wastes constituted by natural elements and thus their coloring is the same of when their are found in nature as non wastes elements. For the wood this color is brown and for hay is ocher/light brown.

Among the classes that show dominant colors in the black and white scale there are two of them that exhibit darker set of dominant colors. These two are represented by tubes and tires. This result confirms what we observed during the annotation process, since both tubes and tires were typically of a dark grey color inside the images in which we found them.

About the remaining classes of wastes again the results confirmed the observations done while annotating, but still there is an interesting consideration to do on the class of dumpsters. This class typically should be colored since during the observations we found them to be colored. Even though the explanation is not immediate to understand the reason lies in the fact that dumpster are open and in most cases are full of scattered waste, thus getting their dominant colors. Otherwise when they are empty their inside can be almost black because of the shadow projected from their walls. Both these scenarios perfectly explain the dominant colors found for this class.

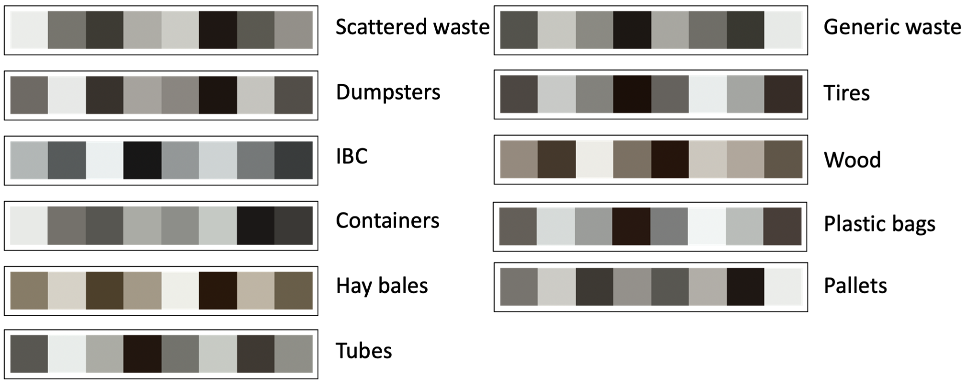

Figure 5.20: Color analysis of the wastes class.

5.2.4 Quality differences in suspicious sites lists

As introduced at the beginning of this section we evaluated a total of 1529 images, among these images: 1036 were from the lists of suspicious sites published online and 493 were from the lists provided by ARPA.

From the lists found online we have created campaigns that have been annotated on images from Mapbox provider (scenario [a]), instead from the lists provided by ARPA we have created specific campaigns that used the orthophoto images for the annotation process (scenario [b]).

Starting from the assumption that every site of these lists should always contain an illegal landfills or at least recognizable wastes we can consider as goodness metric for the lists the acceptance percentage of the sites they contain.

The site images annotated in scenario [a] yield an acceptance percentage of 54%, with 564 accepted sites and 472 rejected sites.

The site images annotated in scenario [b] yield an acceptance percentage of 88%, with 436 accepted sites and 57 rejected sites.

These results confirmed the assumption made in section 3.1.1, that initially was only based on the inspection of the data contained inside of them, about the difference of quality between the two list sources.

These very different acceptance results are not only explained by the quality difference of the processes that realized the lists but also from other two facts. First, the lists found online are clearly realized over a longer time span, making the reliability of the observations more susceptible to possible evolution that is not taken into account by these lists. Second, we do not know from which data sources these online lists have been realized consequently we cannot exploit the same potential given by ARPA lists that have been produced and check on the same image source that we used for our annotation process.

This last fact points out the necessity to have a strong relationship between the lists, from which we start annotating, and the images sources on which we annotate.

5.2.5 Quality comparison of the image sources

In this work, as mentioned in Chapter 3, two different sources of images were employed.

Between the images of these two sources there is a noticeable quality difference, as it is shown in Figure 5.21 with two adjacent images that represent the same area of 117m per edge for both sources.

In this figure it is evident that the images from the orthophoto source has a better quality. This is possibly due to two reasons: a better resolution of the images and a better preprocessing during their realization.

A better resolution allows to have inside the images objects with more sharpened edges and this is effectively confirmed by the fact that orthophoto images have a resolution of 0.2m per pixel compared with the 0.3m per pixel of Mapbox imagery.

A better preprocessing, that in this scenario could be represented by the atmospheric correction of the images, could make the coloring of the images more uniform. This is just what is possible to see in the comparison between the two images, thus confirming also this idea.

This quality difference confirms the necessity of having high resolution images and to make good preprocessing of those, like in the case of the images from orthophoto sources, if the interest is to use them to detect small wastes.

With reference to our case scenario of wastes identification it is possible to show which is the impact of this quality difference during the annotation process.

In Figure 5.21 it is also possible to see two red squared polygons that both represent the same area of ≈4x4 meters.

Inside the polygon on the right, referred to the orthophoto source, it is clearly possible to detect that there are wastes inside of it and also to recognize what these wastes are: a very small concentration of scattered wastes and some pieces of tubes.

Inside the polygon on the left, referred to the Mapbox source, it is only possible to see something white that has a different coloring compared to the surrounding ground.

From this example of visual observation it is clear that from orthophoto images it is possible to recognize even wastes of ≈4x4 meters while on the Mapbox images this is not possible at all.

Figure 5.21: Quality comparison between Mapbox (left) and Orthophoto (right) sources. In these images there are also two squared polygons containing wastes, collocated in the same geographical position in both the images.

In Figure 5.22 it is possible to see a comparison of the annotation counts we have done in images from Mapbox source and Orthophoto source. From this histogram there are two evident differences, between the two image sources, that are related to the classes of: scattered wastes and generic wastes.

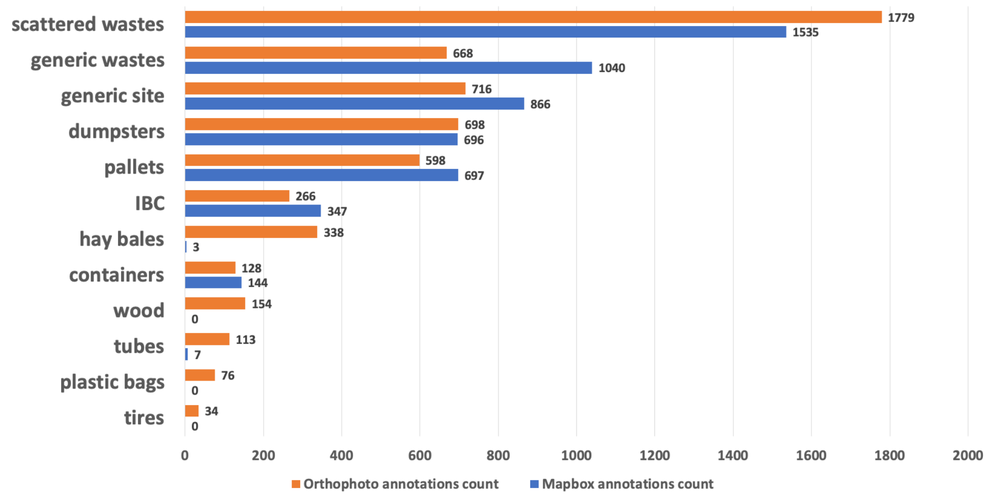

Figure 5.22: Comparison of the annotations count obtained using Mapbox images and Orthophoto images.

In the case of scattered wastes it is possible to see that in orthophoto images the count is ≈16% higher than the one with Mapbox images. This can be explained with the increased resolution and quality of the orthophoto images that allows to better distinguish the waste from the underlying soil. In the case of generic wastes it is possible to see that in orthophoto images the count is ≈36% lower than the one with Mapbox images. Also in this case a possible explanation for this decrease is given by the increased quality of orthophoto images that allows to recognize better the elements that are stacked on the ground classifying them with their correct class instead of using the generic wastes class.

Another difference between the two image sources, already anticipated in section 4.3.2, is that in images from Mapbox source it possible to find visual defects.

An example of this issue can be seen in Figure 5.23 where it is shown an image that we considered not good to be used for our purposes of annotation. All the images that presented similar visual defects like the one in figure were discarded during the annotation process.

A potential explanation of this issue is given by the methods used by Mapbox to create its images. Mapbox relies on other providers as source of the satellite images from which it elaborates its own ones.

To create its own images it merges together different layers that comes from the original satellite sources. Doing this way there is the possibility that different layers, concatenated together to create a larger map, come from different temporal moments or received a different type of preprocessing. One last possibility is that Mapbox has not available, for all the territories that it covers, layers with the same resolution thus concatenating layers with different spatial resolution.

In all these evaluated cases the sharpness and coloring difference between two different adjacent layers is clearly noticeable, like in the example figure.

Figure 5.23: Example of image from Mapbox provider with visual defects

Chapter 6

Conclusions and Future Work

In this thesis we aim to provide tools to investigate the issue of illegal landfill. To such purpose we provide tools for the generation of a dataset that can be used for the development and validation of new methods to identify illegal landfills using remote sensing imagery.

We made a state of the art analysis of the available methodologies for illegal landfill identification to understand which are the most effective also analysing their points of strength.

From this initial analysis we decided to point towards the usage of remote sensing imagery as main source of data and from similar studies in the field we found viable the possibility to apply deep learning methodologies to them.

We carried out an analysis of the different typologies of wastes, commonly found inside illegal landfills, and we used this knowledge to create a dataset that characterize them.

To create such dataset we developed specifics tools for the retrieval of satellite and aerial images and their preprocessing with which we produced high quality georeferenced images usable to create annotations on them.

To the scope of creating the mentioned dataset, an innovative web annotation tool with support to georeferenced data and collaborative work has been developed. The tool was created for the annotation of wastes but also generalized to other georeferenced scenario given that is up to the user the decision of which categories and objects will be annotated given a specific campaign.

With this tool we performed an analysis of the most common waste typologies which led to the identification and characterization of 12 wastes classes inside more than 1500 evaluated images from which it was possible to create a dataset of more than 10900 annotations total.

The analysis of the generated dataset was performed and it let us understand better the illegal dumping issue highlighting which are the most commonly found wastes along with their size and territorial characteristics. Also how different types and quality of image sources can impact on the annotation process.

6.1 Future works

In this section we present the future works that will improve the tools realized in this project.

6.1.1 Annotation tool limitations

This web crowdsourcing tool has been developed alongside the development of the requirements of the project, trying to follow and fulfill as much as possible the needs that came up in the development. For this reason there is still room for improvements voted to remove some current limitations and better generalize the tool in order to make it available for other scenarios.

6.1.2 Generalization to non-geographical campaigns

One of the major limitations that currently the web application has is the possibility to work with geolocalized data only. The whole application is highly tuned towards the use of maps and images that come along with geographical data describing their geoboundaries (that will be used to do geographical reconstruction of the coordinates of the point drawn by the workers in the map with the Leaflet.Draw plugin). A possible future expansion work could be to allow the creation of non-geographical campaign allowing thus the annotation of images that do not have geographical reference attached to them. This would allow to annotate images for other purposes still annotating them with the same methodology of classification and/or annotation (selecting for each drawn annotation the corresponding labels associated to it). This kind of modification will mean mainly a change in the page that allows the creation of campaigns in order to accept an input different from the list of coordinates and in the pages with maps to remove them. The plug-in used for annotation already provides the functionality to obtain the drawn items using pixels coordinates. Also changes in the backend server to manage a more generalized kind of data, still allowing the creation and execution of geographical campaigns.

6.1.3 More customization in creation phase

In the creation phase it is possible to specify the distance and image source only between two hardcoded values. Distance for example can be 0.5km and 1km, it would be interesting to let the admin have more control over it, giving the possibility to choose an arbitrary distance at least for those campaigns created upon Mapbox images (for the campaign created upon orthophoto it will still exist the constraint of having the necessary data to create the image, before having a total coverage of a certain area removing it will not be possible).

Customizable classification questions

Another current limitation in the selection stage is that the question are hardcoded in the frontend structure. For the current project this is not an issue since the interest is to let the worker do a screening stage before actually annotate only the images that are already classified as relevant. However in a future implementation, maybe even generalized to non-geographical campaigns, it would be interesting and useful to let the admin decide for any campaign the classification questions that he requires relevant for that specific campaign. This feature would make even more useful the usage of the selection stage alone by itself without a posterior annotation stage, making the tool a little be more oriented on image classification rather than in recognizing something inside the image.

6.1.4 Implement new image sources

For what concerns the choice of the image sources it is possible to choose between Mapbox (map provider) and images coming from orthophoto preprocessing. Adding more sources would be a useful way to enrich data created from the web application since the same image from different provider could result in different annotations/results (sometimes because of quality difference between providers that allows to spot more or less details or because of different image sources are from different moments in time).

Bibliography

[1] Katharine Glanville and Hsing-Chung Chang. "Remote Sensing Analysis Techniques and Sensor Requirements to Support the Mapping of Illegal Domestic Waste Disposal Sites in Queensland, Australia". In: *Remote Sensing* 7 (Oct. 2015), pp. 13053–13069. DOI: 10.3390/rs71013053.

[2] Jasravia Gill et al. "Detection of waste dumping locations in landfill using multi-temporal Landsat thermal images". In: *Waste Management & Research* 37 (Jan. 2019), p. 0734242X1882180. DOI: 10.1177/0734242X18821808.

[3] Enrico Cadau et al. "SIMDEO: An integrated system for landfill detection and monitoring using EO data". In: July 2013, pp. 3305–3308. ISBN: 978-1-4799-1114-1. DOI: 10.1109/IGARSS.2013.6723534.

[4] Chinatsu Yonezawa. "Possibility of Monitoring of Waste Disposal Site Using Satellite Imagery". In: 2009.

[5] Angel Luis Lucendo-Monedero, Rosa Jorda-Borrell, and Francisca Ruiz-Rodriguez. "Predictive model for areas with illegal landfills using logistic regression". In: *Journal of Environmental Planning and Management* 58.7 (2015), pp. 1309–1326. DOI: 10.1080/09640568.2014.993751. eprint: https://doi.org/10.1080/09640568.2014.993751. URL: https://doi.org/10.1080/09640568.2014.993751.

[6] S. Ren et al. "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.6 (2017), pp. 1137–1149.

[7] J. Redmon et al. "You Only Look Once: Unified, Real-Time Object Detection". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 779–788.

[8] Akshay Dabholkar et al. "Smart Illegal Dumping Detection". In: Apr. 2017, pp. 255–260. DOI: `10.1109/BigDataService.2017.51`.

[9] J. Deng et al. "ImageNet: A large-scale hierarchical image database". In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. 2009, pp. 248–255.

[10] Tsung-Yi Lin et al. "Microsoft COCO: Common Objects in Context". In: *Computer Vision – ECCV 2014*. Ed. by David Fleet et al. Cham: Springer International Publishing, 2014, pp. 740–755. ISBN: 978-3-319-10602-1.

[11] T. Splajt, G. Ferrier, and L. E. Frostick. "Application of ground penetrating radar in mapping and monitoring landfill sites". In: *Environmental Geology* 44.8 (2003), pp. 963–967. DOI: `10.1007/s00254-003-0839-5`. URL: `https://doi.org/10.1007/s00254-003-0839-5`.

[12] OpenEI by the National Renewable Energy Laboratory. *Remote Sensing Techniques*. `https://web.archive.org/web/20200509131625/https://openei.org/wiki/Remote_Sensing_Techniques`. Accessed: 2020-05-09.

[13] Kamil Faisal, Mohamed Alahmad, and Ahmed Shaker. "REMOTE SENSING TECHNIQUES AS A TOOL FOR ENVIRONMENTAL MONITORING". In: Aug. 2012.

[14] Giancarlo Biotto et al. "GIS, multi-criteria and multi-factor spatial analysis for the probability assessment of the existence of illegal landfills". In: *International Journal of Geographical Information Science* 23.10 (2009), pp. 1233–1244. DOI: `10.1080/`

13658810802112128. eprint: https://doi.org/10.1080/13658810802112128. URL: https://doi.org/10.1080/13658810802112128.

[15] Miroslav Kubasek and Jiri Hrebicek. "Involving Citizens into Mapping of Illegal Landfills and other civic issues in the Czech Republic". In: June 2014.

[16] R. Girshick et al. "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation". In: *2014 IEEE Conference on Computer Vision and Pattern Recognition*. 2014, pp. 580–587.

[17] Ying Wang and Xu Zhang. "Autonomous garbage detection for intelligent urban management". In: *MATEC Web of Conferences* 232 (Jan. 2018), p. 01056. DOI: 10.1051/matecconf/201823201056.

[18] Syed Waqas Zamir et al. "iSAID: A Large-scale Dataset for Instance Segmentation in Aerial Images". In: *CoRR* abs/1905.12886 (2019). arXiv: 1905.12886. URL: http://arxiv.org/abs/1905.12886.

[19] Centre for Remote Imaging Sensing Processing. *Optical Remote Sensing*. https://web.archive.org/web/20200509132452/https://crisp.nus.edu.sg/~research/tutorial/optical.htm. Accessed: 2020-05-09.

[20] John Rogan. "Remote sensing for mapping and monitoring land-cover and land-use change—An introduction". In: *Progress in Planning - PROG PLANN* 61 (May 2004), pp. 269–279. DOI: 10.1016/S0305-9006(03)00064-3.

[21] S Silvestri and M Omri. "A method for the remote sensing identification of uncontrolled landfills: formulation and validation". In: *International Journal of Remote Sensing* 29.4 (2008), pp. 975–989. DOI: 10.1080/01431160701311317. eprint: https://doi.org/10.1080/01431160701311317. URL: https://doi.org/10.1080/01431160701311317.

[22] Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton. "ImageNet Classification with Deep Convolutional Neural Networks". In: *Neural Information Processing Systems* 25 (Jan. 2012). DOI: 10.1145/3065386.

[23] Lu Yang et al. "Parsing R-CNN for Instance-Level Human Analysis". In: *CoRR* abs/1811.12596 (2018). arXiv: 1811.12596. URL: http://arxiv.org/abs/1811.12596.

[24] Dan Claudiu Ciresan et al. "Deep Big Simple Neural Nets Excel on Handwritten Digit Recognition". In: *CoRR* abs/1003.0358 (2010). arXiv: 1003.0358. URL: http://arxiv.org/abs/1003.0358.

[25] Kaiming He et al. "Deep Residual Learning for Image Recognition". In: *CoRR* abs/1512.03385 (2015). arXiv: 1512.03385. URL: http://arxiv.org/abs/1512.03385.

[26] Kaiming He et al. "Mask R-CNN". In: *CoRR* abs/1703.06870 (2017). arXiv: 1703.06870. URL: http://arxiv.org/abs/1703.06870.

[27] M krishna et al. "Image classification using Deep learning". In: *International Journal of Engineering & Technology* 7 (Mar. 2018), p. 614. DOI: 10.14419/ijet.v7i2.7.10892.

[28] Piotr Dollar. *Learning to Segment*. https://research.fb.com/blog/2016/08/learning-to-segment. Accessed: 2020-05-11.

[29] Qiangqiang Yuan et al. "Deep learning in environmental remote sensing: Achievements and challenges". In: *Remote Sensing of Environment* 241 (2020), p. 111716.

[30] Xiao Xiang Zhu et al. "Deep learning in remote sensing: A comprehensive review and list of resources". In: *IEEE Geoscience and Remote Sensing Magazine* 5.4 (2017), pp. 8–36.

[31] Liangpei Zhang, Lefei Zhang, and Bo Du. "Deep learning for remote sensing data: A technical tutorial on the state of the art". In: *IEEE Geoscience and Remote Sensing Magazine* 4.2 (2016), pp. 22–40.

[32] Nina Ødegaard et al. "Classification of ships using real and simulated data in a convolutional neural network". In: *2016 IEEE Radar Conference (RadarConf)*. IEEE. 2016, pp. 1–6.

[33] Rocio Nahime Torres et al. "Mountain summit detection with Deep Learning: evaluation and comparison with heuristic methods". In: *Applied Geomatics* (2019), pp. 1–22.

[34] Ari Silburt et al. "Lunar crater identification via deep learning". In: *Icarus* 317 (2019), pp. 27–38.

[35] Christopher Lee. "Automated crater detection on Mars using deep learning". In: *Planetary and Space Science* 170 (2019), pp. 16–28.

[36] Wenwen Li and Chia-Yu Hsu. "Automated terrain feature identification from remote sensing imagery: a deep learning approach". In: *International Journal of Geographical Information Science* 34 (Nov. 2018), pp. 1–24. DOI: 10.1080/13658816.2018.1542697.

[37] GuiSong Xia et al. "DOTA: A Large-scale Dataset for Object Detection in Aerial Images". In: *CoRR* abs/1711.10398 (2017). arXiv: 1711.10398. URL: http://arxiv.org/abs/1711.10398.

[38] Darius Lam et al. "xView: Objects in Context in Overhead Imagery". In: *CoRR* abs/1802.07856 (2018). arXiv: 1802.07856. URL: http://arxiv.org/abs/1802.07856.

[39] Gencer Sumbul et al. "BigEarthNet: A Large-Scale Benchmark Archive For Remote Sensing Image Understanding". In: *CoRR* abs/1902.06148 (2019). arXiv: 1902.06148. URL: http://arxiv.org/abs/1902.06148.

[40] Ritwik Gupta et al. *xBD: A Dataset for Assessing Building Damage from Satellite Imagery*. 2019. arXiv: 1911.09296 [cs.CV].

[41] Ritwik Gupta et al. "Creating xBD: A Dataset for Assessing Building Damage from Satellite Imagery". In: *ArXiv* abs/1911.09296 (2019).

[42] Mang Tik Chiu et al. *Agriculture-Vision: A Large Aerial Image Database for Agricultural Pattern Analysis*. 2020. arXiv: 2001.01306 [cs.CV].

[43] Kevin Sparks et al. "Citizen science land cover classification based on ground and aerial imagery". In: *International Conference on Spatial Information Theory*. Springer. 2015, pp. 289–305.

[44] Rocio Nahime Torres et al. "Crowdsourcing landforms for open GIS enrichment". In: *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE. 2018, pp. 604–611.

[45] Ahmed Loai Ali et al. "Rule-guided human classification of Volunteered Geographic Information". In: *ISPRS journal of photogrammetry and remote sensing* 127 (2017), pp. 3–15.