Don’t let the bark beetles bite! Detecting bark beetle infestations and monitoring the health of forests through remote-sensing analysis of satellite images

Marko Horatio Mekjavic

4th Year Project Report
Computer Science
School of Informatics
University of Edinburgh

2022

Abstract

Detection of bark beetle infestations in early stages is a vital task for efficient management of forests to minimise potential economic losses and mitigate their further spreading. Remote sensing methods are becoming ever more important as an objective approach to enable monitoring health of forests and detect potential bark beetle infestations early on. This project aimed at developing a change-based model for analysing satellite data in the visual and near infra-red light spectrum and recognising potential bark beetle infestations that could help local forest rangers in monitoring larger forest areas in a more efficient manner.

Developed model is capable of recognising changes in the forest area with a success rate of 87.5%. Moreover, it is also capable of estimating the area impacted by the attack with an over-estimation error of approximately 30%. The model also comprises of a fine-tuned cloud-masking component that achieved a score of 90.6%, which allows for it to perform advanced pre-processing on the data. Henceforth, in comparison to other theoretical solutions, the developed model can analyse even images with cloud impurities. Developed using Python programming language adds to its potential for being easily deployed as an assistance tool for the forest rangers to monitor health of forests in their communities.

Besides the model, the project has also created and openly released two new data-sets that can be used for future research projects. The first consists of pre-processed data for 24 areas located across Slovenia and Croatia which have recorded bark beetle attacks. Second data-set consists of more detailed information, specifying the exact amount of land impacted by the attack in the Milanov Vrh region. These are also the first openly available prepared data-sets recording bark beetle attacks, that can be found online.

Research Ethics Approval

This project was planned in accordance with the Informatics Research Ethics policy. It did not involve any aspects that required approval from the Informatics Research Ethics committee.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Marko Horatio Mekjavic*)

Acknowledgements

To professors for their time, family for their support, and classmates for their help.

Table of Contents

Chapter 1

Introduction

Global warming is becoming an ever more serious problem humanity is facing in the 21st century. In the last decade we have seen the previously unimaginable consequences of our actions in various forms of natural disasters.

1.1 Motivation

Coming from a small country of Slovenia, I had the opportunity to develop an appreciation for forests from an early age. Slovenia is an incredibly small country, one of the smallest in Europe. However, its geographic position, consisting of the Julian Alps, Mediteranean seaside, as well as the Panonian Basin in a radius of just a 100 kilometers, makes it a natural phenomena. This diversity can be seen in the various different ecosystems that exist side by side, thanks to vast areas being covered by forests and other vegetation, which play a vital role in preserving and maintaining local ecosystems. From a young age we are taught in schools the importance of forests in preserving our local as well as global ecosystems. However, in the last decade due to global warming we have seen how these previously very different local ecosystems are disappearing. Extreme weather conditions in recent years such as big intra-day temperature changes and unpredictable hailstorms have not only impacted us humans, but also our surroundings, resulting in large areas of forests being damaged. But what is more, these broken trees have become perfect breeding grounds for a small dangerous species, the bark beetle. These small organisms when in large populations, can cause unimaginable damage and permanently change the forest and consequently the local ecosystems. As there is no current way of successfully fighting against the beetle attacks, there is a lot of emphasis on preventing such breeding grounds in the first place or isolating the attacked trees. As my parents own a small forest, I have seen how hard it is to efficiently manage the forest even as small as ours. The feeling of being powerless has motivated me to try and utilise the tools of modern technology to try and find a solution that could help local communities monitor the health of their forests and prevent large scale bark beetle attacks by identifying and locating potential breeding areas or already damaged regions.

1.2 Project Goals

The main objective of this project was to develop an algorithmic solution that can help local communities detect and prevent bark beetle attacks, by utilising the knowledge gained here at The University of Edinburgh throughout my four years.

To clarify further, the aim of this project was to provide local communities with a tool that would analyse large areas of forest and locate damaged areas, by cutting down the time and costs associated with the current methods used to monitor forests. This way the local communities can prioritise which regions of forest they should inspect first, cutting down the time needed to find damaged regions and increase their overall efficiency. For the solution to be helpful to all communities regardless of their available budgets, it was important the solution uses publicly available data.

1.3 Summary of Results

The results of this project can be divided into two main points, successful development of an algorithmic solution for recognising bark beetle infestations, and development of two new data-sets that can be utilised by the research community.
Firstly, a change-based model for detecting bark beetle infestations through analysing satellite images has been successfully developed and tested. On a limited data-set the model achieved 85% success rate in identifying regions that had been impacted by the bark beetle infestations. Moreover, it also achieved a 87.5% success rate in recognising change in the forest area, be that negative or positive. In terms of its performance of quantifying the size of the area impacted by the bark beetle attack, the algorithm over-estimated the impacted area by approximately 30%, which now serves as a baseline for any future detection algorithms for this given data-set. The model also comprises of a cloud masking component, which has been fine-tuned and achieved an *f-score* of 90.6%.
Secondly, two new data-sets were created and openly released for the community to use. The first data-set contains information about 24 areas that had been impacted by the bark beetle attack, spread across eight regions in Slovenia and Croatia. For each region two satellite images have been obtained and pre-processed, one prior and one after the attack. The second data-set contains a detailed information of how much land had been impacted by the bark beetle attack in the region of Milanov Vrh. This can serve as a performance measure for future detection algorithms.

The code and data for this project are also openly available: `https://github.com/Pompey21/Don-t-Let-the-Bark-Beetle-Bite-You`.

Chapter 2

Background and Literature Review

2.1 The Beetles

Bark beetle is one of about 6000 species in the beetle subfamily of *Scolytinae*. Even though as the name suggests, not all members of this family indulge in the lifestyle of feeding in the inner bark *(phloem)* layer of trees, but rather feed in fruit and seeds, or tunnel into non-woody trees *(herbaceous)* plants. [39]

Figure 2.1: Cartoon depicting a beetles band. [26]

2.1.1 What are the Bark Beetles?

There are many different known species of bark beetles, many of which are not particularly harmful to forests as they only feed off the dying and unhealthy trees and hence importantly contribute to the natural cycle of forests by helping with decomposition of dying trees. However, due to only a handful of bark beetle tribes, this important role they play is usually overseen and overshadowed by their potential damaging effects. To non-entomologists the bark beetles are especially known as pests and harmful insects.

Figure 2.2: Bark Beetle. [12]

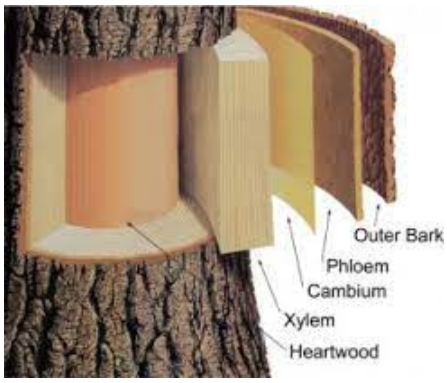


Figure 2.3: Structure of a tree. [23]

Some tribes, such as European elm bark beetle *(S. multistriatus)* and the American elm bark beetle *(Hylurgopinus rusipes)* are known especially due to their involvement in spreading and transmitting the Dutch elm disease fungi *(Ophiostoma)*, which can severely damage elm forests. On the other hand, the tribes of mountain pine beetle *(Dendroctonnus ponderosae)*, southern pine beetle *(Dendroctonus frottalis)* in North America, and spruce ips *(Ips typographus)* in Europe, are known as major pests as they can severely damage healthy conifer trees by drilling into them and cutting their water supply. [39]
Bark beetles are tiny organisms, averaging approximately 5 millimeters in size, which makes them incredibly hard to spot in the wild. While some beetles breed in only one species of trees, there are some that can feed off almost any tree. As mentioned, bark beetles often attack trees that are already weakened by disease, overcrowding, con-specific beetles, or another physical damage. [39]

2.1.2 Life Cycle

Bark beetles go through four stages of life: egg, larvae, pupae, and adult - similar to the one of a butterfly. The time spent in each stage depends on the species as well as the temperature of their environment, with warmer temperatures decreasing the times. In the UK, the beetle has a rather long cycle, ranging from 12 to 18 months. It generally begins when adults bore into a tree and lay their eggs in the phloem of the tree, which usually occurs at the end of Summer. Once the eggs hatch, the larvae then live in the tree, feeding on the living tissues bellow the bark, often leading to death of the tree by cutting the water supply. At the end of the larvae stage chamber-like structures are constructed under the bark which serve as a place for the pupae to overwinter until they are ready to emerge as adults, which then repeat the process.

2.1.3 T(rees) - Day

Bark beetles are usually not damaging to the healthy trees, as they tend to attack already weakened and dead trees. However, as explained some families tend to be more

Figure 2.4: Chamber-like structures (1). [33]

Figure 2.5: Chamber-like structures (2). [25]

aggressive and can develop large populations, which can pose a serious threat even to the healthy trees. Healthy trees can respond with producing sap, resin or latex, in response to an attack, which kills the imposter. However, when the population of beetles increases significantly, the trees can easily get overwhelmed and hence cannot protect themselves from the beetles, which are therefore known as pests. [18]
Global warming in recent years has also had an effect on the success of the bark beetles. Increase in temperature has allowed the bark beetles to expand into new areas, previously unfamiliar with the beetles. Moreover, warmer conditions are also more favourable for beetles to breed, resulting in much larger populations. However, the most crucial addition to the equation are the consequences of extreme weather conditions that we observed in recent years. Floods, big intra-day temperature changes, colder winters, and hail storms, can severely damage the trees and hence reduce their abilities to defend against the beetles. Broken and weakened trees along with warmer temperatures present ideal conditions for the beetles. Once their populations grow significantly, they can pretty much attack any tree and overwhelm their immune system. [11]
Bark beetles attack the trees by drilling through their bark and into the inner layers. Once the beetles successfully drill their tunnels deeper into the tree, they create a complex systems of tunnels that can eventually cut the trees' water supply, preventing the them from successfully supplying water to their leaves. This causes trees to gradually start dying. Firstly, the leaves change colour from green to orange, and afterwards they turn grey, when the tree is completely dead.

2.1.4 How to Recognise Infestations

There are different methods that can be used for recognising bark beetle infestations in forests, however they can be divided into two groups. The first group consists of methods that can only be executed from the ground and require us to be located next to the tree, while the second allows for utilisation of remote sensing tools.
Trees that have been attacked can be distinguished by recognising the presence of boring dust and pitch tubes on the outside of the bark, which the beetles drilled into the tree and hence produced the dust. Moreover, on the inside the bark beetle adults and larvae can be spotted. However, it is important to note that these characteristics are sometimes rather hard to recognise and demand more focus. [15]
Another way of recognising bark beetle attacks is by observing the trees' crowns. A change in colour reports an attack taking place. While the orange/brown colour indicates

earlier stages when the tree is still defending, the grey colour marks defeated trees. However, the second mentioned technique sadly has a couple of limitations. A change of colour only occurs when the beetle has already made serious progress and had severely damaged the tree. [8]

Figure 2.6: Change in colour of ever-greens' crowns as a consequence of a bark beetle attack. [19]

2.2 Measures in Place

As mentioned, the beetles become a major threat once their population significantly increases in size. Therefore, the most successful methods of fighting against their attack, is to spot their presence quickly, or preventing them from increasing in size, by influencing the factors that can lead to creation of ideal breeding conditions.

Bark beetles are incredibly small organisms and are quite hard to spot with a naked eye, which makes them incredibly hard to fight against. Their presence in the forest is usually spotted once the evergreen tree crowns had already changed their colour. As explained above, the change in colour suggests the tree is already decaying and will slowly die. It is important to spot these changes early on, so that infected trees can be cut down, removing potential breeding grounds for the beetles. Otherwise, when the infected regions are already significantly sized, many possible actions of fighting against the pest become redundant as the population of the beetles is simply too large to control and eradicate.

Therefore, it is important to try and prevent the spread of bark beetles by monitoring the state of the forest and making sure that weakened trees are removed on time and that forests are frequently being sampled and tested so as to spot any potential attack on time. This however, has been done in a very inefficient way. Many organisations resort to hiring rangers and forest managers that monitor the health of the forest manually, by walking through the forest and spotting potential problems and damages. This method is of course very inefficient because it is sometimes unfeasible to expect the whole region to be thoroughly inspected on time. Another potential way of monitoring the state of

forests is by using the airplanes to fly over the forest and take high-resolution images and try spot the regions that need further examination on ground. However, this is an incredibly expensive method which many communities cannot afford. Moreover, when it comes to large territories of forest, such as in Canada and Asia, even this method becomes inefficient. [17]

Recent technological advancements have opened new opportunities, such as utilisation of satellite- and drone-gathered data with remote sensing tools to spot potential breeding grounds and infestations in the forest. The applications and efficiency of this approach is currently still being explored.

2.3 Satellites and Remote Sensing

As it was previously stated in the *Introduction*, the main objective of this project was to explore the opportunities of utilising publicly available and accessible satellite data for the purposes of development of an algorithm that can help monitor the health of forests, namely locating and identifying the damaged areas of forests. There are many satellite programmes available that offer access to such data, however, the Copernicus programme was chosen due to its incredible characteristics.

2.3.1 Remote Sensing

Remote sensing refers to the process of detecting and monitoring the physical characteristics of an area by measuring its reflected and emitted radiation from a distance. This is usually achieved using different sensors and high precision cameras. Examples range from sonar systems on ships used to create images of the ocean floor, to cameras attached to airplanes and satellites taking images of large areas of Earth's surface. Both examples allow us to obtain large amounts of data in a much more efficient way comparing it to the difficulties of achieving that manually, being on the ground or in the ocean. [34]

In the last decade we have witnessed a big increase in its applications in various different sectors, including forestry. A wide range of spectral and spatial options now exist for the user to tailor their information needs with the most appropriate data inputs. Image preparation techniques, such as radiometric and geometric adjustments which are now taken for granted, previously occupied a significant amount of total time spent on forest characterisation and application projects. However, with the development of new data delivery options that allow for the user to request these image analysis to be already completed for a minimal price or even for free, has truly pushed the applications of remote sensing in various different fields. [41]
Moreover, services like Sinergise [30] and Mundi Web-Services [31] now also provide the satellite data formats that are easily analysed and manipulated, such as *numpy* array and *pandas* dataframes for Python programming language.

2.3.2 Copernicus Mission

Copernicus is the European Union's Earth observation programme, developed and managed by the *European Space Agency* (ESA). [3]
The programme provides information services regarding different Earth observations. The data is collected by numerous sensors, which are positioned on the ground, at sea or in the air. The satellite programme is only one of the ways the data is collected. [4]

2.3.2.1 Sentinel Satellites

Sentinel mission is part of the Copernicus mission and comprises of a network of satellites. The first satellite from the Sentinel family was the Sentinel-1A, launched in 2014. The Sentinel family can be further divided into six groups. Each Sentinel mission is based on a constellation of two satellites to fulfil revisit and coverage requirements, providing robust data-sets for Copernicus services. [5]

- ***Sentinel-1*** is a polar-orbiting, all-weather, day-and-night radar imaging mission for land and ocean service, mainly used for monitoring changes in Earth's crust.
- ***Sentinel-2*** is a polar-orbiting, multi-spectral high-resolution imaging mission for land monitoring, providing imagery of vegetation, soil and water cover, inland waterways and coastal areas.
- ***Sentinel-3*** is a multi-instrument mission measuring sea-surface topography, sea- and land-surface temperature, ocean colour and land colour with high-end accuracy and reliability.
- ***Sentinel-4*** is a payload devoted to atmospheric monitoring that will be embarked upon a Meteosat Third Generation-Sounder (MTG-S) satellite in geostationary orbit.
- ***Sentinel-5*** is a payload that will monitor the atmosphere from polar orbit aboard a MetOp Second Generation satellite.
- ***Sentinel-5 Precursor*** is the forerunner of Sentinel-5 to provide timely data on a multitude of trace gases and aerosols affecting air quality and climate.

As mentioned each Sentinel mission comprises of at least two satellites, which translates in providing images and data for each region every 5 to 10 days, depending on the mission.
The satellite mission of interest to this project was the *Sentinel-2* mission, which offers high-resolution imaging of land, including vegetation, soil, coastal and other water covered areas.

2.3.2.2 Copernicus Sentinel-2

The Copernicus Sentinel-2 mission is based on a constellation of two identical satellites in the orbit. Each satellite carries an innovative wide swath high-resolution multi-spectral imager with 13 spectral bands for measuring various different aspects of the Earth's surface, ranging from absorption of *Short-Wave Infra-Red* to *Coastal Aerosol*. The full list of bands and their respective characteristics is shown in *Figure* 2.8 bellow.

Figure 2.7: Showing the Sentinel family consisting of all six mission. [28]

[6]

The first satellite, Sentinel-2A was launched on 23rd of June 2015, while the second satellite, Sentinel-2B, was launched on 7th of March 2017. With the launch of the second satellite, the revisit time of each region, was shortened to only 5 days. Another impressive characteristics offered by the satellites is their resolution, which for some of the bands is just 10 meters. This means that each pixel represents an area of 10 square meters. This resolution is very impressive for a publicly available satellite programme. Comparing it to the *Landsat* programme operated by the *United States Geological Survey* which offers 30 meter resolution. [6]

2.4 Detection of Plants and Vegetation

Live green plants that contain chlorophyll, absorb solar radiation in the photo-synthetically active radiation spectral region, which ranges between 400 and 700 nanometers. More specifically, they tend to absorb mostly the red and blue light from the visual spectrum while the green light gets reflected. What is more, leaf cells also tend to re-emit solar radiation in the near-infrared spectral region, as the photon energy at wavelengths longer than 700 nanometers is too small to synthesize organic molecules, which are created during photosynthesis. Henceforth, while observing the plants through the red light lenses, the plants appear black as they absorb the large portion of energy. On the other hand, while observing them from the lens of a near infra-red light, they appear to be very light, as this energy largely gets reflected.
There are several different spectral indices, which are particularly good at recognising

Sentinel-2 bands	Sentinel-2A		Sentinel-2B		
	Central wavelength (nm)	Bandwidth (nm)	Central wavelength (nm)	Bandwidth (nm)	Spatial resolution (m)
Band 1 – Coastal aerosol	442.7	21	442.2	21	60
Band 2 – Blue	492.4	66	492.1	66	10
Band 3 – Green	559.8	36	559.0	36	10
Band 4 – Red	664.6	31	664.9	31	10
Band 5 – Vegetation red edge	704.1	15	703.8	16	20
Band 6 – Vegetation red edge	740.5	15	739.1	15	20
Band 7 – Vegetation red edge	782.8	20	779.7	20	20
Band 8 – NIR	832.8	106	832.9	106	10
Band 8A – Narrow NIR	864.7	21	864.0	22	20
Band 9 – Water vapour	945.1	20	943.2	21	60
Band 10 – SWIR – Cirrus	1373.5	31	1376.9	30	60
Band 11 – SWIR	1613.7	91	1610.4	94	20
Band 12 – SWIR	2202.4	175	2185.7	185	20

Figure 2.8: Sentinel-2 bands and their respective characteristics. [38]

vegetation and separating it from other man-made constructions. For this project the *Normalised Difference Vegetation Index (NDVI)* and *Reg-Green Index (RGI)* were chosen for recognising the vegetation.

Studies conducted by the *Myneni et al*[14] have shown that the *NDVI* is directly related to the photosynthetic capacity and hence energy absorption of plant canopies.
Building on top of that, unhealthy trees and the ones containing little number of leaves, will be reflecting much more of the red and absorbing more of the near infra-red light, this can be observed in *Figure* 2.9. If we further elaborate, regions of forest that have been damaged and are as a consequence either less densely populated or the leaves have started to change colour, will henceforth be observable on the satellite data when calculating the *NDVI*.
The *NDVI* is calculated form the *Red* and *Near Infra-Red (NIR)* wavelengths with the following relation:

$$NDVI = \frac{NIR - Red}{NIR + Red}$$

Where *Red* and *NIR* stand for the spectral reflectance measurements acquired in their respective regions. As it can be observed, the *NDVI* takes upon the values in range between -1 and +1, which makes it especially useful for analysis in comparison to other optional ratios between the visible and infra-red light spectrum.
From the equation we can now observe that in the case of the healthy plants, where the refelctance of near infra-red light is high and the one of the red light is low, the calculated *NDVI* index will tend to take positive values closer to 1. On the other hand in case of an unhealthy plant, where less red light as well as less near infra-red light will be reflected, the calculated *NDVI* index will tend to take smaller values.

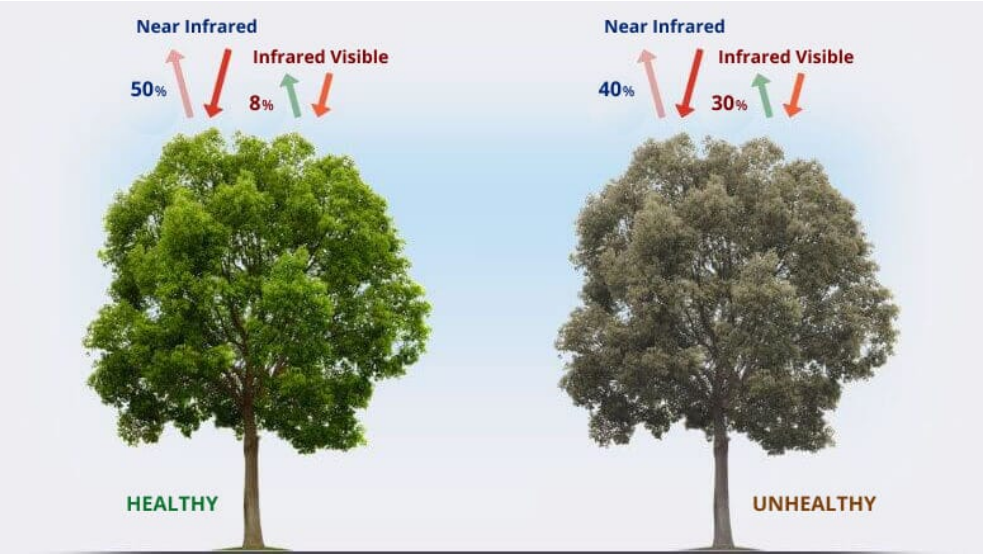


Figure 2.9: Normalised Difference Vegetation Index for healthy and unhealthy trees. [35]

The *Red-Green Index*, or *RGI*, is another simple graphical indicator, which can be used for recognising and separating vegetation from other types of land covers. This method is much simpler as it does not involve the near infra-red spectrum, where it is easier to separate vegetation. Henceforth, for classification of the vegetation alone it may not be as good of an indicator as *NDVI* is. As the name suggests, it is only measuring the ratio between the red and green light absorbance and can be calculated in the following way:

$$RGI = \frac{Red}{Green}$$

As explained, the evergreen conifers start changing colours of their crowns as the bark beetle attack progresses. The colours change from green to brown, orange, or red. Henceforth, this should result in a decreased absorbance of the red light and an increased absorbance of the green light. Therefore, this index could be useful for identifying potentially infected regions. This method was also evaluated by *Barta et al*[9] showing *RGI* achieving formidable results in identifying late-stage attacks.

2.5 Research Development so Far

As a response to increased bark beetle attacks, there has been a lot of research and development conducted in how various different technological advancements can be used to fight against bark beetle infestations. These range from analysing satellite data, to using drones and planes with high resolution cameras to scan the large forest areas. Study conducted by *Barta et al* titled *Comparison of Field Survey and Remote Sensing Techniques for Detection of Bark Beetle-Infested Trees* [9] has compared the efficiency of traditional forest surveying for detection of bark beetle infestations with a modern approach of using unmanned aerial vehicles - or drones - equipped with high resolution cameras and sensors. The drones scouted the forest area by flying over it, allowing

them to take pictures, which could later be analysed. The study successfully concluded, that remote sensing techniques, such as change in red-green light spectrum, could be efficiently used to detect infected trees, especially in the later stages of the attack. Another study conducted by *Stafanova et al* [27] has taken the approach of utilising drones a step further. The Spanish-Russian research team explored the opportunity of applying deep learning techniques for analysing images collected by the drones, classifying different stages of the attack the trees had undergone. This was achieved by focusing on the shape, texture and colour of tree crowns.

Both of the mentioned studies have achieved phenomenal break-through and have proven that the current technology available can be successfully applied in detecting bark beetle infestations. However, both studies used a very high precision equipment, which had a spatial resolution of less than a meter, allowing for an individual tree to be seen in the pictures. Moreover, the studies focused on identifying individual trees in a smaller forest sample. While the results were quire impressive, the limitation of this approach definitely lies in limited scalability opportunities. For the drone to capture good quality pictures it needed to fly rather close to the ground and have ideal weather conditions. This means that it is still unrealistic to expect that larger field areas can be covered with drones in an efficient time, especially considering that weather conditions will be far from ideal.

In order to be able to monitor larger areas of forest the idea of utilising satellite data has been explored by various research teams. While results obtained by the studies mentioned above showed that bark beetle infestations can be detected from aerial images, the new question arised, whether it would be possible to achieve similar results on a larger scale. Covering larger areas of forest could be achieved by utilising satellite data, however, it offers much lower spatial resolution. In comparison with the aerial observations obtained by drones, satellite images offer a spatial resolution between 10 to 60 meters depending on the programme and the measuring system used.
A research team lead by *D. F. Gomez* [20] has explored the abilities of different publicly available satellite programmes in detecting bark beetle infestations in the south-eastern parts of the United States. In their study they mainly explored the opportunities of using *MODIS* and *Sentinel-2* obtained data. The conclusion of the study showed that *Sentinel-2* achieved much better results, as it provided data with much higher resolution.

In order to observe and understand different stages of infestations, a study conducted by *Barta et al* [10] tracked their development by observing the changes recorded in different light spectra obtained by the *Sentinel-2* satellites. The observations showed that the biggest changes were seen in the near infra-red and red light absorbance.
This research break through was taken even further by *Angel Fernandez-Carrillo et al* [16], who lead a team focusing on developing algorithmic models for detecting potential infestations. They used the *Sentinel-2* obtained data-set and developed two models based on change detection approach, where they tracked development of the forest region through time. However, besides utilising the *Normalised Difference Vegetation Index*, which measures the relation between the red and near infra-red light, as the main measure of change, they also managed to construct several advanced biotic damage products, that could keep track of the state of the forest. The results obtained showed impressive capabilities of the algorithm, confirming that remote sensing tools

for analysing satellite data can be used for identifying infestations.

However, all the mentioned studies had the luxury of having their own testing sites, which allowed them to obtain true field data. These were then used to construct a new data-set from the data obtained through the *Sentinel-2* mission. Developing and thoroughly testing such algorithms is otherwise much harder. Hence, the implementation of these techniques and algorithms into reality can be much harder.
One such example was the study conducted by *Kranjcic et al* [21], which tried to develop a more simplistic algorithmic approach for detecting and visualising the aftermath of a large bark beetle attack in the Čabar region in the northwestern part of Croatia. Since the region was far less well recorded by the local rangers than that used in the study by *Angel Fernandez-Carrillo et al* [16], advanced biotic damage products could not be developed efficiently. However, the team decided to utilise *Normalised Difference Vegetation Index* and *Red-Green Index*, since they can measure the change in red, green and near infra-red light spectrum, which were identified as spectra with biggest observed change during the bark beetle infestation. The developed algorithm was used to classify trees in the region accordingly.

Chapter 3

Methodology

3.1 Development of Idea

3.1.1 Initial Idea

Initial idea behind the project was to use a machine-learning approach for the development of a model capable of recognising potential bark beetle attacks from the data obtained through the publicly available satellite programmes. While the research advancements mentioned in *Research Development so Far* (2.5) predominantly offered theoretical solutions, the aim of this project was to develop a solution that could easily be implemented and used by the local communities. One of the limitations of the previously mentioned research advancements was the usage of the off-the-shelf geographical information systems, such as the *QGIS* software. While these programmes offer user-friendly interface that can easily be used, they do not allow much of the customised data-analysis but rather offer only in-built functionalities. Henceforth, a more programming-like approach where the methods and models would be developed from scratch provided an opportunity for more customised solution and a product that could be eventually easily deployed. Such approach would also allow for more different variables to be observed by the model at once, where instead of considering only bands 2, 3, 4 and 8, we could introduce new bands. This more programming-like approach would allow us to explore if there is another combination of variables that can produce more favourable results. Moreover, instead of using the already available methods that can be used in the *QGIS* software, we could perform more advanced fine-tuning of parameters offered by more powerful machine learning libraries. In order to execute this idea, the first step was obtaining a labelled data-set.

3.1.2 Data-sets Available

Preferably the data-set would provide exact location of which parts of the forest had been impacted by the bark beetle attack. Since each pixel on the image obtained by the *Sentinel-2* satellite represents an area of 10 square meters, it was assumed that individual pixels where the attack occurred would be labelled. Such format of the labelled data could easily be divided into training and testing sets used by the classification algorithm.

However, the reality showed that the data-sets were much different from the desired format. Since there was no publicly available data-set found online, decision was made to reach out to different research teams, that had been conducting research in this area, as well as forest services and other organisations that have been dealing with bark beetle infestations. As a result, three different data-sets were obtained.

The Forest Service of the Republic of Slovenia records information about the regions that had been impacted by the bark beetle attacks. Every region contains a report, specifying the log of all operations conducted in the region, such as planned felling or other type of intervention. However, the data does not quantify the impact of the bark beetle attacks nor did it specifically label the pixels with higher density attacks. The regions were mapped as polygons that were estimated by the local forest rangers.

The second data-set was provided by the Colorado State Forest Service, similarly to the previously mentioned one, it contained regions where the attack had occurred. These were mapped as polygons and were displayed on an interactive map of Colorado. [29] In comparison to the first data-set, it did not provide any additional information.

The last data-set obtained, was provided by the research team from *University of Zagreb*, lead by *Prof Kranjcic*, which had been used for their study on the regional bark beetle attacks.[21] Data-set was focusing on a region of Čabar, which was much smaller than the ones covered by the previous two data-sets. Besides regional information regarding the attacks, the team also provided more detailed information of an area called Milanov Vrh, for which it was recorded exactly how much land had been impacted by the bark beetle attack. However, this particular data-set for the Milanov Vrh region had not been used in the study conducted by the team. [21]

3.1.3 New Idea

As mentioned, the limitation preventing the execution of the initial idea of developing a machine learning model, was the lack of data in the specified labeled format. Hence, the initial idea had to be modified based on the available data. Since the available data only provided rough estimations of the regions where the attack occurred, the machine-learning approach was replaced with an idea of implementing change-based detection algorithm, similar to the approach taken by the *Fernandez et al* [16]. This way satellite data of a region prior to the attack can be compared to the one obtained after the attack. With this approach we can avoid the need for having a detailed labelled data-set. Instead, the only required data is the one specifying the location of bark beetle attacks, which can even be an estimation in terms of the area where it was located as well as the size of the impact. That is because the change-based model would compare the results of an image obtained after the attack to the ones of an image obtained prior to the attack. If we know that the region had undergone a bark beetle attack, then we can estimate the algorithm's performance in terms of being able to successfully detect regions that had been attacked. Of course, having some quantitative estimator explaining the severity of the impact would allow us to test the limitations of the algorithm much better. The algorithm may be very successful at detecting areas where the beetles impacted more than a third of a forest, but is not very good at identifying regions that had been impacted less.

In comparison with the study by *Fernandez et al*[16] the available data-sets were much less detailed. Their data-set was built internally by collaborating with the Forest Service responsible for monitoring and taking care of their forest, which they had used for many of their studies. Years of well-documented records have provided them with advanced data which allowed them to create and build much more detailed data-sets. This allowed them to develop more advanced biotic damage indicators, which take into account different aspects such as humidity and acidity of different regions of the forest. Instead, the data-sets we obtained only consisted of the information where certain bark beetle attacks occurred. These regions were solemnly an estimation of the attack, which of course offered much lower accuracy and precision. However, these were the real-life records that Forest Services had been collecting for years and are much more realistic in terms of what one can expect from such organisations to have at their disposal. Having an ideal testing set is something that will be very hard to find in real-life especially if we want to create a tool that can be deployed and used by these organisations.
Since we did not have the necessary data for creating even simple biotic damage indicators, we decided to utilise other remote sensing tools, focusing on the ones which we can use for analysing satellite data obtained from the *Sentinel-2* mission. The damage to the forest caused by the bark beetles can most easily be observed through the change of the colour of the evergreen trees, that is why we aimed at exploring different opportunities of how to detect such change in satellite images. Initially it was suspected that the changes could be easily observed by analysing satellite images in their true colours, however, looking at *Figure* 4.1 we can see that it is rather hard to spot any difference between the image prior and the one after the attack. Therefore, we started exploring other colour-estimation mechanisms that are used in remote sensing.

Inspired by the study conducted by *Barta et al* [9], we looked closer into applications and capabilities of simple colour detection mechanisms. While we initially suspected that only visible light spectrum will be used in our analysis, we discovered that the near infra-red light spectrum was also a good measuring tool for indicating the change in forest's health after the attack. The study done by *Barta et al* [9] has shown that the biggest change observed from the satellite data was that recorded by the *Red-Green Index* and *Normalised Difference Vegetation Index*. Observing both measuring tools with a higher focus, decision was made to try and implement both indices into our change-based detection model.
Both indices were also used in the study conducted by *Kranjcic* [21] for visualisation and estimation of damage in the region of Čabar. The study focused on using unsupervised learning for the classification of pixels and did not focus providing any concrete results obtained from testing as it was mainly focusing on visualising potential impact of the attack in the region. This project would therefore aim at developing a model that can also be more thoroughly tested and its performance can be evaluated.
Another improvement that we aimed at was the development of more advanced pre-processing methods. While majority of the studies explained how they were limited by the availability of cloudless satellite data, we aimed at developing an algorithm that can perform analysis also when the satellite images are not in the ideal shape. For that reason, we wanted to focus on developing a cloud detection and masking component, which could be integrated into the solution. Implementing the mentioned component and developing the algorithm in a more programming like approach where the methods are

build from scratch, truly provides an opportunity for the solution to be more applicable to every-day usage by local communities. For once, it is considerably more easily deployed than other models that have been developed using the *QGIS* platform, and secondly, the cloud masking component allows the algorithm to perform on non-ideal data which is something that we should account for in real-life applications.

From the previously mentioned data-sets, two were chosen; the one provided by the Slovenian Forest Service, and the last one, which was provided by the Croatian research team. The main reason behind this decision was two-fold. The regions covered by the two mentioned data-sets were much more easily identified on the map comparing to the ones mentioned in the data-set provided by the Colorado Forest Service. Data for the infected areas in the Čabar region were provided in coordinates, and hence locating them was no problem. On the other hand, the regions provided by the Forest Service of Slovenia did not provide specific coordinates, however, with the help of provided descriptions and a detailed interactive map [24], it was not hard to locate the regions using the `Google Earth Engine` platform [13] and obtain rough estimations of their exact coordinates. Regions form the Colorado were predominantly far away from any settlements and were hence a bit problematic to find and identify.
Another reason for choosing the data-set provided by *Kranjcic et al* [21] was also their more detailed record of the Milanov Vrh region, for which they had exact numbers in terms of how much land had been impacted by the bark beetle attack. This would allow to obtain a more concrete estimation of the algorithm's performance as it allows for exact comparison. Moreover, this data-set has previously not been used in any of the studies and by using it here, a new perspective can be brought to the research community.

3.2 Data

The main objective of this project was to use publicly available satellite data. There are many different satellite programmes, offering free access to past data, however, *Sentinel-2* mission from the *Copernicus* programme was chosen as it offered most suitable characteristics.

3.2.1 Platforms Used

The data from the *Copernicus* programme can be accessed through various different platforms, ranging from simple API calls, to more user-friendly in-built code editors. Throughout this project there were two main platforms used for accessing data, the `Google Earth Engine` [13] for easy exploration and visualisations, and the `Sentinel Hub` Python API provided by Sinergise [30] for performing analysis on images and development of the algorithm.

3.2.1.1 Google Earth Engine

`Google Earth Engine` [13] is a geo-spatial processing service, which provides free access to a multi-petabyte catalog of satellite imagery and geo-spatial data-sets from

different satellite programmes, including *Sentinel-2*. It provides an `Earth Engine` API, which is available both in Python and JavaScript. Moreover, it also offers an in-built interactive web-based code editor with real-time visualisation of data for JavaScript. This makes the platform very useful as the images can be positioned on the World map while simultaneously allowing the user to play around with different geometrical shapes and other modifications. However, while the visualisation is extremely efficient, the limitation of the platform is the hardly accessible raw data. Moreover, the web-based interactive code editor also has a limited access to JavaScript libraries for data analysis.

3.2.1.2 Sentinel Hub

`SentinelHub` is a Python package which provides access to the various data-sets obtained by the *Sentinel* satellite mission. Users can open a trial account for a duration of one month, offering free access to almost any satellite data collection from the *Sentinel* programme. [32]
The package allows retrieval of any supported format of satellite data, ranging from a normal `PNG` image formats, to more advanced `TIFF` formats. Moreover, the package allows for any data to be easily converted to a format suitable for data analysis, such as numpy arrays. However, the limitation of the platform is that it only offers the retrieval of the information in a format suitable for analysis, any sort of advanced visualisation options, such as the ones provided by the `Google Earth Engine` are not supported. [32]

3.2.2 Data Format

As mentioned before, the *Sentinel-2* satellite records an image of an area of approximately 290 square kilometers. These larger swaths are later broken into smaller tiles measuring approximately 100 square kilometers. The raw data is then prepared and formatted to the extent it can be easily accessed by the user, which includes many geoforencing methods. The following two main products are generated by the mission: [1]

- **Level-1C** - provides Top-Of-Atmospheric (TOA) reflectance
- **Level-2A** - provides Bottom-Of-Atmospheric (BOA) reflectance

The main difference between the two product types is the clarity of the images, where Level-2A offers a better clarity. This is achieved through running advanced algorithms in order to perform an atmospheric correction, which accounts for the error associated with the electromagnetic energy passing through the atmosphere from the satellite to the Earth's surface and back. This results in images being much clearer and with a higher contrast between the colours.

It is important to note, that all the Level-2A data is essentially produced through the pre-processing procedure of the Level-1C data. The data dating back from May 2017 has already been pre-processed and can be available in a better format of Level-2A. Desiring to use older data, the users can perform their own pre-processing of the Level-1C data by utilising the `Sen2Cor` software developed and released by *ESA*. [2]

Since the data for this particular project dated further back than May 2017, decision was made to attempt performing the pre-processing analysis of the available Level-1C data. After familiarising with the software's documentation, there were two possible methods identified for performing the transformation of the data. One included manually creating the environment for the `Sen2Cor` script in the terminal, while the other included downloading the `SNAP` software for satellite image analysis, into which the `Sen2Cor` script could be implemented. [2]
Implementation of both methods resulted in attempting to debug many programming errors that were brought to the user's attention. However, after exploring the matter further and consulting the forums, it was pointed out that the current software update is not compatible neither with the Mac nor Linux operating systems which were available at the time of the project. Therefore, a decision was made to use the Level-1C data instead as it not needed any further customisation.

3.2.3 Region of Interest

The main geographical focus of this project were the regions provided by the Slovenian Forest Service as well as the region of Čabar, which data was provided by the Croatian research team. All together accounting for 7 different regions. These were then further divided into smaller areas, resulting in 24 different areas.
All the areas used in this project are located between the altitudes of 500 to 1700 meters above sea level. They are predominantly covered with various conifer trees, sometimes also mixed-tree forests. This made these regions particularly useful for the project, as conifers are predominantly evergreens, meaning that any colour change in their crowns is something unusual. Moreover, all the chosen regions were intentionally located far away from any city or settlement to reduce the possible error of the algorithm falsely identifying them as infected forest regions.

All the regions can be seen in *Figure* 3.1 .

More about the data used for the project can be found in *Chapter 4*.

3.3 System Design

The new idea behind the algorithm was to make a decision based on a recorded change that had been detected. For that, two images for each area needed to be obtained, one prior and one after the attack. These would then be pre-processed to the extent that the *Normalised Difference Vegetation Index* and the *Red-Green Index* could be easily calculated for each pixel on both images. The pixels, which would be marked by both indices would be classified as damaged and unhealthy forest areas. Once the algorithm classified pixels on both images, the number of indicated pixels would be compared between the two pictures. This way a change in forest could easily be recorded.

The algorithm was broken down into individual components, each responsible for one of the functionalities of the algorithm:

- **Controller** - receives the input indicating the region of interest, and calls upon other components to perform their functionalities. At the end of the execution, it

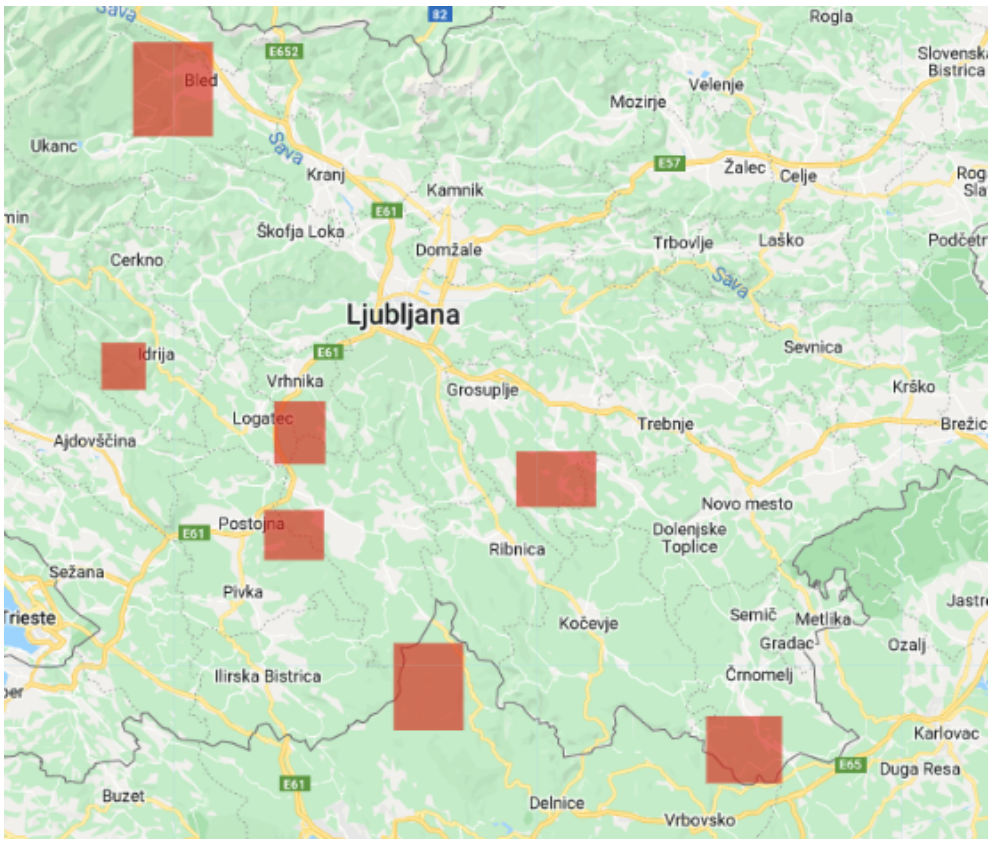


Figure 3.1: Regions of Interest shown on the map.

returns the analysed satellite images and the calculated results back to the user.

- **Data Provider** - responsible for obtaining the satellite data for the desired region of interest. It also calls upon the *Pre-Processing* component before returning the data.
- **Pre-Processor** - calls the *Cloud Masking* algorithm, responsible for pre-processing the data, which is then returned to the *Data Provider*.
- **Detector** - calls two algorithms, *Normalised Difference Vegetation Index* and *Reg-Green Index*, for calculating the respective indices and then classifies the pixels into healthy and non-healthy ones. It then returns both images and the calculated results to the controller.

The explained components can be seen in *Figure* 3.2 bellow.

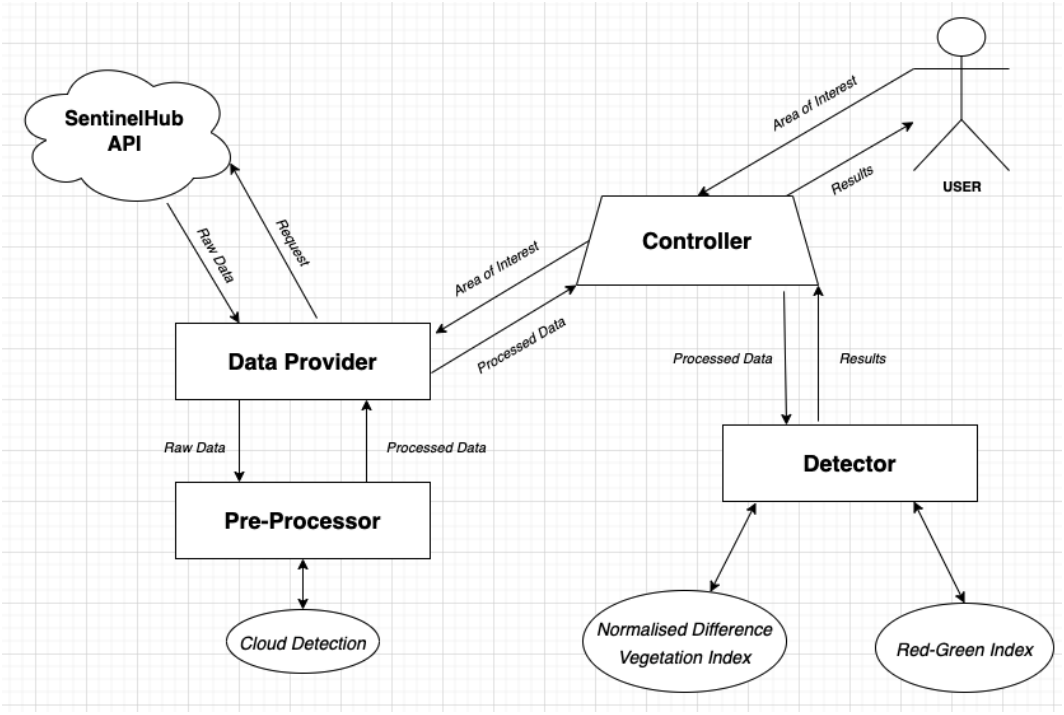


Figure 3.2: System Design of the Model.

The cloud-shaped object depicts *SentinelHub* API, while the oval ones are representing algorithms responsible for individual tasks. There are also three components, *Data Provider*, *Pre-Processor*, and *Detector*, which are responsible for connecting various different components, hence they were depicted as rectangles. The *Controller* was depicted as a trapezoid, while the user is drawn as a stick figure.
Communication between different components can be divided into two groups; a simple call to an algorithm is depicted with a double headed arrow, while communication between other components is depicted with a single arrow with an explanation as to what is being transferred between the units.
Each of the mentioned algorithms is also explained in more detail in its respective subsection bellow.

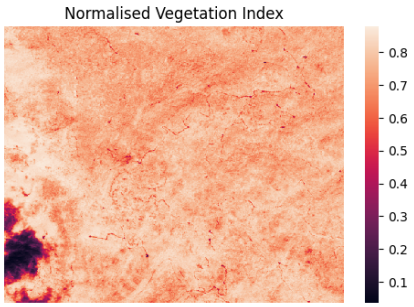


Figure 3.3: Calculated NDVI for each pixel.

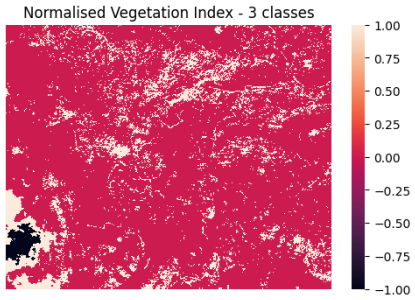


Figure 3.4: Classified results, where white and black represent damaged trees.

3.3.1 Normalised Difference Vegetation Index

Sentinel-2 mission provides data in the form of exactly 13 bands. One of which, band 4, only records the absorbance in the *red* visual light spectrum, while band 8 records absorbance in the *near infra-red* light spectrum. By accessing the data through the *SentinelHub* API, the measured data for each image could easily be translated into a 13-dimensional numpy array, where every band was represented by one array. Although not all bands are able to operate at the same spatial resolution, luckily both band 4 and 8 are able to provide the same spatial resolution of 10 meters. Hence, calculating the *NDVI* index was a straight forward calculation involving the two matrices.
Once each pixel was calculated the *NDVI* value, as seen in *Figure* 3.3, the next step was to classify them into healthy and unhealthy plants. In order to determine the cutoff value of the *NDVI* which would indicate the boundary between healthy and unhealthy trees, the study conducted by *Xiao et al* [42] was used as a reference point. The study included approximately 9000 subjects belonging to 215 different species. It concluded that setting the cutoff point to be the *NDVI* value of 0.7 yielded a 88% accuracy. Henceforth, all pixels with the values above or equal to 0.7 were classified as healthy while the rest as unhealthy. Results can be seen in *Figure* 3.4.

However, as it can be observed from the picture, there are some limitations. Man-made constructions, such as forest roads will be classified as unhealthy trees and even some regions of forest with lower tree density may be falsely classified. Moreover, clouds are also recognised as damaged trees.

3.3.2 Red Green Index

In terms of visible light spectrum, *Sentinel-2* provides three bands measuring *blue*, *red* and *green* light. As it can be observed from *Figure* 2.8, all three bands offer 10 meters of spatial resolution.
By accessing the data through the *SentinelHub* API, the measured data for both bands could be easily translated into two respective numpy arrays. Calculating the index then became a trivial operation including a simple array division.
Once each pixel was calculated the *RGI* value, the next step was to classify them into healthy and unhealthy plants. After consulting the research team from the *University of Zagreb*, a cutoff was determined to be set to 0.6. Healthy trees would score lower since

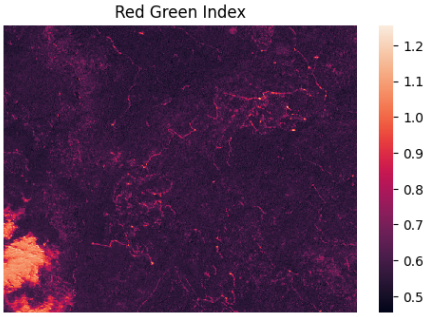


Figure 3.5: Calculated RGI for each pixel.

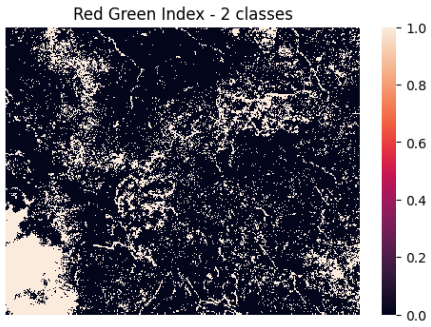


Figure 3.6: Classified results, where white pixels represent damaged trees.

they would reflect more green and less red light.
RGI calculated values can be seen in *Figure* 3.5, while the classification results can be seen in *Figure* 3.6.

3.3.3 Cloud Masking

In many respects clouds are a phenomena, but in the World of remote-sensing they are very much a problem as they prevent access to the real data we wish to observe. One of the limitations of using publicly available satellite data is its sparsity. Quite often a region of interest was fully or partially covered with clouds. Obtaining a completely clear image is almost impossible as it requires a lot of luck for having no clouds in the sky.
The first idea was to simply ignore the clouds and analyse the picture as a whole, assuming that *NDVI*. and *RGI* indices will not detect the cloud, however, as it was observed from *Figures* 3.4 and 3.6, clouds do get classified as unhealthy trees. Henceforth, developing a cloud recognising component was crucial.

When researching what advancements have been done in this area, many state-of-the-art cloud masking algorithms are predominantly based on data obtained from satellite missions containing a temperature measuring sensor tool, which makes the cloud detection much easier. Namely, because clouds tend to be cooler than the Earth's surface, while wavelength reflection can be more similar to those of snow or other water vapour like objects.
After more thorough research, there were 3 dominant cloud-masking algorithms identified that were based on *Sentinel-2* or *Landsat 8* data-set collection, hence fitting the criteria of not having the availability of the temperature measuring component. *Sen2Cor* which was developed by the European Space Agency using *Sentinel-2* data, *FMask* developed by the United States Geological Survey using *Landsat-8* data, and *MAJA* developed by the French Center National d'études Spatiales. [36]
Initially it was decided to try and utilise *Sen2Cor* algorithm, however, as explained in *Data Format* section, the current publicly available version of the algorithm is not compatible with Mac and Linux operating systems, which were at our disposal. Hence, the other two algorithms were explored in more detail. *FMask* had a major limitation of being trained on *Landsat 8* data-set, which meant a much lower spatial resolution of

30 meters, comparing to 10 meters one from the *Sentinel-2*. On the other hand, *MAJA* algorithm has not yet been expanded and implemented in a format which could be used for this project.

After some more research, another algorithm named *s2cloudless* developed by Sinergise was found, which is based and was developed using *Sentinel-2* data specifically, hence ensuring the 10 meter resolution. [7]

3.3.3.1 s2cloudless algorithm

As opposed to the previously mentioned algorithms, *FMask*, *Sen2Cor*, and *MAJA* which implement a set of static or dynamic thresholds for cloud detection, the *s2cloudless* algorithm opts for a different approach, utilising the advancements in the field of machine learning. It was explained by the research team lead by *Zupanc et al* [7] how the model was firstly trained to recognise each pixel individually, assigning a certain probability of whether it is covered by a cloud. Hence, the pixel's probability of representing a cloud does not depend on neighbouring pixels. These are only taken into consideration when constructing the cloud mask of a given scene from its cloud probability map, by performing various morphological operations, such as convolution.

The *s2cloudless* algorithm is a pre-trained machine learning model which, given a satellite image in the raw `TIFF` format, estimates which pixels are belonging to the area covered with clouds and assigns a probability to each pixel. The algorithm is readily available through the Python library that can easily be installed through the *pip* command. The pre-trained model `S2PixelCloudDetector` takes as an input Sentinel-2 image of shape (n, m, 13) (all 13 bands) or (n, m, 10) (bands 1, 2, 4, 5, 8, 8A, 9, 10, 11, 12) and returns a raster binary cloud mask. The classifier can instead of a raster cloud mask return a cloud probability map of shape (n, m), where each pixel's value is bound between 0 (clear-sky-like pixel) and 1 (cloud-like pixel). User can control cloud probability threshold and/or post-processing steps - such as convolution and dilation. The following are parameters the user can fine-tune according to its preferences:

```
param threshold ::
Cloud probability threshold. All pixels with cloud probability above
threshold value are masked as cloudy pixels. Default is 0.4.

param all_bands::
Flag specifying that input images will consist of all 13 Sentinel-2
bands or just 10.

param average_over::
Size of the disk in pixels for performing convolution (averaging
probability over pixels). Value 0 means do not perform
this post-processing step. Default is 1.

param model_filename::
Location of the serialised model.
```

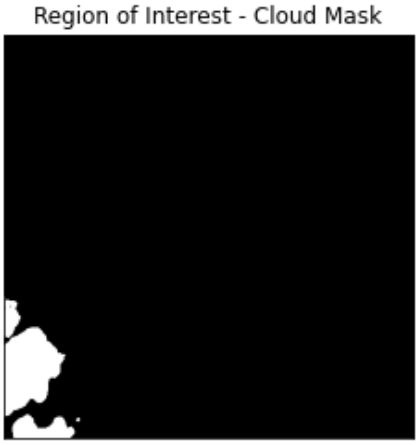


Figure 3.7: Calculated cloud mask.

Figure 3.8: Cloud mask applied to the original image.

`If None the default model provided with the package is loaded.`

[22]

3.3.3.2 s2cloudless fine-tuning

There were a couple of parameters that had to be fine-tuned in order to optimise the performance of the `s2cloudless` algorithm. Initially the idea comprised of utilising the `Scikit-Learn`'s `GridSearchCV` method, however, due to the format of data, decision was made to write an independent *precision* and *recall* calculator. This was especially useful for our concrete example as the classes are heavily imbalanced and hence the two indicators can be very helpful in determining what combination of parameters performs best. More on fine-tuning of the cloud-masking algorithm can be found in section 4.1.

The chosen parameters were the following:

- `threshold` - 0.2
- `average over` - 4
- `dilation size` - 2
- `all bands` - *False*

Results obtained after generating a cloud mask can be seen in *Figure* 3.7 and *Figure* 3.8.

Chapter 4

Experiments

In this section we shall go over different experiments which were constructed for the developed model and evaluate the results. A major limitation in constructing efficient and thorough tests had been the lack of any publicly available data in the format of precise coordinate representation of parts of forests that were severely damaged during the bark beetle attacks. Majority of the data available comes in reports, where local communities and authorities report the attacks occurring in certain regions. These reports are more of an estimation by the local forest rangers rather than thoroughly collected data. As mentioned in the *Region of Interest* (3.2.3) subsection, there were 8 regions across Slovenia and Croatia, further divided into 24 areas, for which enough data was collected. These are also the regions that have been used for testing.
There were three different experimental tests conducted on the developed algorithm. Firstly, the cloud-masking component of the algorithm was tested as an individual component in order to obtain best performing combination of parameters. The second set of tests focused on exploring algorithm's performance in recognising damaged forest areas. Lastly, the algorithm was tested on how well it estimates the damage made by the bark beetle attack.

4.1 Cloud Masking Algorithm

As stated before, the *Cloud Masking Algorithm* was developed following a tutorial and a documentation provided by Sinergise. However, due to my modifications and lack of experimental results, I decided to conduct independent experiments in order to quantitatively estimate its performance. The results provided by their research team [7] were very much limited only to a handful of images and hence did not provide a solid estimation and evaluation of the algorithm's performance. Moreover, after using the algorithm myself, I noticed the presented results were not as realistic as they claimed, at least for my examples.
Quantifying the cloud-masking algorithm's performance was essential as the algorithm responsible for detecting changes in forest covered areas, is not capable to distinguish between a damaged forest area and a cloud, as seen in *Figures* 3.4 and 3.6. Observing the performance of the cloud-masking algorithm, I noticed that the algorithm was

extremely efficient in recognising larger clouds or those with higher density, which indeed are a dominant sort of clouds. However, the algorithm did not perform very well when the clouds were less dense and at times translucent, meaning you can see through but with a blur. These clouds are also known as *cirrus* clouds and in contrast with ordinary clouds tend to appear in higher altitudes and are less noticeable. Cirrus are usually formed as warm, dry air rises, causing water vapor deposition onto rocky or metallic dust particles at high altitudes. In the region of Čabar, which was my main region of interest, these clouds are more common due to geographical layout, where the warm and cold air tend to mix frequently. [40]

4.1.1 Study Area

The experiment was limited to the study regions mentioned in *Region of Interest* (3.2.3) section. The area chosen for this experiment is a cropped region of the Čabar, with the following coordinates specifications:

```
bounding box = [14.48200,45.54980,14.54738,45.59538]
```

The first two numbers represent the coordinates of the bottom left most point, while the later two are the coordinates of the bottom right most point of the bounding box. All coordinates are taken in the `WGS84 CRS` system [37], which is the global standard.

4.1.2 Testing

The first step was creating labelled data, which would act as a ground-truth against which the results of `S2PixelCloudDetector` algorithm would be compared against. This was accomplished by utilising the `Photoshop` software to manually label all cloud pixels on two images taken of the region mentioned above. The two images were later transformed into simple numpy arrays. Both images measured in size `1129 x 1152` pixels, which in total accounts for 2601216 pixels.
Secondly, a set of randomly chosen parameter combinations for the `S2PixelCloudDetector` was generated. The set included 42 different parameter combinations, listed in subsection 3.3.3.1. For each of the parameter combinations, a classifier was obtained and generated two cloud-masks for both pictures. These were then compared against the manually generated cloud-mask, and their *precision*, *recall*, and *f-measure* were calculated.

4.1.3 Results

The results for the 8 best performing parameter combinations can be observed in the *Table* 4.1.

As we can observe from the table, the highest precision scores were achieved when all bands were observed. However, in cloud masking component *recall* might be a bit more important as it tells us what percentage of all cloud pixels were successfully identified. Of course *precision* is also a very important measure, however, identifying a

Number	Threshold	average	dilation	all bands	precision	recall	f-measure
1	0.2	4	2	FALSE	0.873	0.942	0.906
2	0.6	1	1	FALSE	0.873	0.942	0.906
3	0.8	7	4	FALSE	0.850	0.955	0.899
4	0.4	3	2	TRUE	0.890	0.907	0.898
5	0.7	6	5	TRUE	0.876	0.906	0.891
6	0.2	5	6	FALSE	0.901	0.878	0.889
7	0.5	2	2	TRUE	0.911	0.847	0.878
8	0.6	8	5	TRUE	0.932	0.823	0.874

Table 4.1: Eight best performing parameter combinations for s2cloudless algorithm.

bit more pixels as clouds is not as bad as not recognising those that should be identified as clouds.

The full results list can be found in the *Table* B.1.

4.1.4 Limitations

Obtaining the ground-truth mask was especially hard as various open source resources available lack a comprehensive connectivity and hence any sort of advanced analysis and testing can become an obstacle. In this concrete example the issue was the non-compatibility between the tools that provide formation of geometrical objects on satellite images, such as `Google Earth Engine`, and those capable analysing matrices, such as Python programming language and its associated libraries. Should such methods be available, this would allow for an automated labelling and hence more tests could be performed.
Throughout the process of obtaining the manually generated cloud mask, there were some pixels which were not easily classified even with a naked eye. Nevertheless, these pixels were not as common and definitely out-numbered by other more easily distinguishable.

4.2 Damaged Areas Testing

For the purposes of testing how well the algorithm identifies regions that had been attacked by the bark beetles, a testing data-set was created from the two sets of data provided by the Forest Service of the Republic of Slovenia and the Croatian research team lead by *Professor Kranjcic*, which had been mentioned in subsection 3.1.2 under the *Development of Idea*. The main idea behind the experiment was to test the abilities of the algorithm in identifying the areas of infected forests. There were 24 study areas chosen, which can be grouped in 8 distinct regions located both in Slovenia and Croatia. The mentioned regions can be seen in *Figure* 3.1.

4.2.1 Source of Data

The data was obtained with the help of local forest rangers both in Slovenia and Croatia, as well as the research team lead by *Professor Kranjcic*. The rangers would inspect the forest, and record the region as infected in case bark beetles were observed.
The limitation of the data-set lays in the lack of previously strictly defined criteria, which would had been used across all regions, clearly stating what percentage of trees/forest must be infected by the bark beetle in order for the area to be identified as infected. Henceforth, the data obtained is exposed to the purely subjective perceptions of the rangers as to what they thought was worth noting/recording. Moreover, while discussing the data with one of the senior forester working at the Slovenian Forest Service, it was explained that the infected regions defined from the coordinates provided by the rangers may be exposed to some significant error in terms of the accuracy of the specified location.

4.2.2 Data Format

The data was obtained in a form of a list of regions that had been infected with the corresponding year of the attack, as well as a report for each individual region, stating the reasons for the change in forest. The report provided information regarding any planned felling and specified the location. Each region had been defined by a polygon shown on the interactive map. [24]
Each region from the list was compared against its respective report, filtering out those that had undergone planned felling and had been the subject of the bark beetle infestations in the same year. This way it was insured that man-made changes would not interfere with the study. Initially the technique of masking the regions where the planned felling took place was considered, however, due to difficulties and the limitations of the API used, this idea was quickly abandoned.
Once the regions had been filtered, each region was approximated by a rectangle drawn using the `Google Earth Engine`'s editor. The coordinates of the rectangle were recorded and prepared into a format acceptable by the SentinelHub API.

4.2.3 Experiment

The data-set is exposed to some significant errors, however, for the purposes of this experiment we shall assume the data had not been compromised and will treat it as a ground truth. The idea behind the experiment was to obtain a relatively clear and cloud-less image of the region prior and after the bark beetle attack. The algorithm would then be run on both images and the number of white pixels, the regions that the algorithm identified as potential areas of damaged trees, would be counted. The number of pixels prior to the attack would then be compared to the number calculated on the image after the attack. If a significant change in the number of pixels had been recorded, this would then indicate/confirm that the algorithm had successfully identified a bark beetle attack. In order to objectify this experiment, a significant change had been quantified to be above 30%.
While obtaining the images prior and after the attack, each picture undergone two safety checks to confirm it was clear enough and of the high enough quality to be used in the

experiment. Firstly, the algorithmicly detected cloud coverage was set be bellow 5%. Secondly, the image was manually inspected to ensure its quality. This was especially useful as some images were not flagged by the cloud-detecting algorithm, however the fog produced by the cirrus clouds was strong enough to interfere with the satellite's sensors. Furthermore, it was also ensured that the images contained as little snow as possible, as this would result in pixels being white, which means that the algorithm cannot perform proper analysis on the image.

4.2.4 Results

The results of the whole data-set including all 24 areas can be found in B.1, which is available in the *Appendix B*.
Observing the performance of the algorithm, three distinct patterns were spotted while analysing results. The first one showing an increase in number of infected land, second showing little to no change, and the last one showing a significant decrease. The introduction of the third pattern is especially important as it will help us analyse the performance of our algorithm.

Change in infected area	Number of samples
Increase	17
No Change	3
Decrease	4

The performance of the algorithm will vary depending on how we define the successful identification of an infected region. Namely, should only an increase in infected regions be treated as successfully identifying an infected region?
Observing the four samples where a reverse change had been detected by the algorithm, a simple solution would be to compare images prior to that year. However, due to the limitations of available data, this was not possible. As stated in the 2.3.2.2 section, the *Sentinel-2* mission has been active only from June 2015 and hence any data prior to that is not obtainable.
Since the bark beetle attacks in the mentioned regions, where a decrease in damaged areas was spotted, occurred between 2014 and 2016, it can be assumed that the spotted decrease in damaged areas was indeed the consequence of forests healing and restoring from an attack. Note that in the reports for each region there is no listing of any other potential causes resulting in damaged areas. Nevertheless, algorithm's performance was calculated against different classifications. The first including the recorded decrease as a successfully identified damaged area, while the second excludes all four samples from the calculations.

Set	Success Rate (in %)
Including	87.5
Excluding	85.0

As it can be observed from the table above, the algorithm successfully identified 87.5% of the infected regions or exactly 85% if samples with decrease in damaged areas are not included.

4.2.5 Limitations

There were many limitations and problems that had been encountered while conducting this experiment and it is important they be mentioned and explained.

4.2.5.1 Subjectivity of Forest Rangers

The initial error identified was the lack of predefined criteria used by the rangers to classify a region as being infected by the bark beetle. One of the problems this technique created was putting regions with different percentage levels of infection into the same bucket. This really created an unfair testing set as the algorithm may be very good identifying regions with higher concentrations.

4.2.5.2 Diversity in Regions

While testing the algorithm's performance on different regions, another limitation was spotted. The chosen data-set for testing is very diverse in terms of types of regions, ranging from flat areas densely covered with forests, to mountainous areas where forests are less dense. This resulted in algorithm detecting different number of white pixels, damaged areas, for different regions. Luckily the experiment was planned in a way that the number of detected pixels on an image was compared directly to the same region but prior to the attack. This in a way decreased the importance and impact of the diversity in the data-set. However, it is important to note that images of mountainous regions are harder to analyse for various reasons.
For one, the forests are much less dense, which when using satellite data with only 10m resolution can become an issue. Secondly, the area can be covered by snow for longer periods of time, which results in pixels being much whiter which is not ideal when making analysis since the algorithm uses specific colour bands for its analysis. Lastly, the higher mountain peaks create shadow, which in contrast to snow results in darker pixels.

4.2.5.3 Approximation of Regions

As it was explained in 4.2.2 each region was later approximated by drawing a rectangle around the region using `Google Earth Engine`'s code editor, which allowed for drawing polygons and calculating their coordinates. The limitation of this technique was the approximations rather than exact regions stated by the Forest Service of the Republic of Slovenia. One of the main reasons for that was the limitation of not being able to use more complicated polygons when requesting the satellite data from the Sentinel Hub API. This resulted in settling for rectangular approximations of each region. This of course introduced another error, however, as the regions chosen were all remote forests, this error was less significant as the extra land covered in the image was still a forest rather than a city or a settlement.

4.2.5.4 Accessibility of Pictures

While obtaining the satellite images for each of the regions, limitations were spotted with regards to the accessibility of the images. Namely, at times it was hard to obtain an image that met the previously explained criteria: cloud coverage being less than 5% and minimal snow coverage. For some of the regions it was impossible to obtain an image of the desired area for the entire year. This was especially the case for images dating between 2015 and 2017. The primary reason for that is namely the fact that until March 2017, the Sentinel-2 mission comprised of only one satellite, resulting in a smaller number of available images. This problem created an error of introducing longer time difference relating to the images taken after the attack. In case of not being able to obtain an image from the year of a bark beetle attack but instead using an image from a year later, this meant that the forest already had the opportunity to restore and self-heal to an extent, resulting in less accurate results of the intensity of bark beetle infestation.

4.3 Milanov Vrh

The data provided by the Croatian research team also contained a more detailed report on a Milanov Vrh area, for which the forest rangers performed a more thorough investigation, recording the exact amount of land that had been damaged by the bark beetle attack. This allowed for performing a more detailed experiment, testing how well can the algorithm detect the damaged areas by comparing its land estimation to the one recorded by the rangers.

4.3.1 Source of Data

As mentioned, the data-set was provided by a research team from the University of Zagreb. The data was collected manually in collaboration between the research team and the regional forest service. The infected trees were thoroughly marked, from which an area of the infected forest was estimated and recorded.
Luckily, the area did not suffer from any other reasons that could have caused the forest to decrease in size, such as planned fellings. Hence, any recorded change was solemnly assigned to the bark beetle infestations.

4.3.2 Data Format

Milanov Vrh area was broken into smaller sections, and for each the damage created was recorded. Amongst the gathered information is the estimation of the volume of damaged wood (in m^3), which should be cut down, the volume of wood obtained during felling (in m^3), and the estimation of how much forest had been impacted (in *hectares*). The overview of the data can be seen in the table bellow, while the raw format can be accessed in the *Figure* A.1.

Region Name	Marked as infected (m^3)	Area (*hectares*)	Cut down (m^3)
Milanov Vrh	9423.0	366.4	9423.0

4.3.3 Experiment

While the precise bounds of the region covered by the Milanov Vrh was impossible to obtain from the literature, decision was made to utilise the gradient decent calculator using the `Google Earth Engine` tool to help draw a polygon around the study area. In *Figure* 4.1, the region can be observed in true colour composition, both prior (2015) and after (2016) the bark beetle attack. The constructed polygon measured in size approximately 42,903 km^2, which translates to the picture having the size of 630 by 681 pixels.

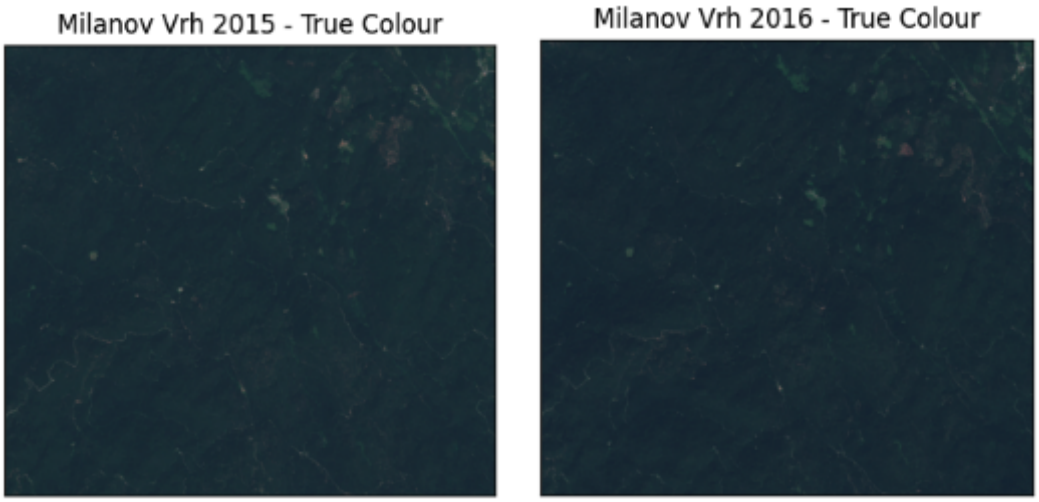


Figure 4.1: Two satellite images of Milanov Vrh area, prior to the bark beetle attack on the left (taken in 2015), and after the bark beetle attack on the right (taken in 2016).

The rectangle polygon approximation of the region is defined with the following coordinates:

```
bounding box = [14.50883,45.56407,14.59570,45.62113]
```

Where the first two decimal numbers represent the coordinates of the bottom-left most point, and the remaining two the top-right most point of the rectangle.

Once the coordinates of the polygon were obtained, they were recorded and passed into the Python script which obtained satellite data using the SentinelHub API. Two satellite images were obtained, one from the 2015, prior to the attack, and one from 2016, after the attack. The dates for the images were selected based on the amount of clouds in the frame, luckily both images contained no cloud coverage. Both images can be seen in *Figure* 4.1, where the left image shows Milanov Vrh in 2015 prior to attack and the one on the right was taken in 2016 after the attack. For a naked eye the two images are almost identical, however, we shall see later on that the algorithm managed to find the differences.

Once the images were obtained, the algorithm was applied to both, following the procedure explained in *System Design* (3.3) section. Once the algorithm pin-pointed the

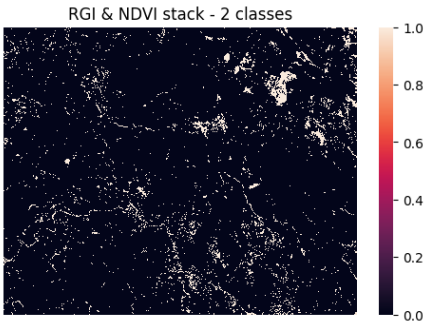


Figure 4.2: Milanov Vrh before attack.

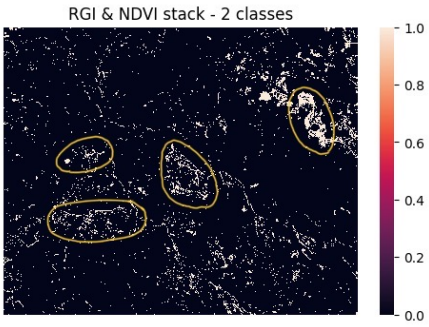


Figure 4.3: Milanov Vrh after attack.

pixels representing the damaged areas of the forest, they were counted and calculated against the whole area.

4.3.4 Results

The results from the experiment can be seen in the table below. The algorithm performed rather well, four major areas of bark beetle infestations were spotted by the algorithm, as shown in *Figure* 4.3, that have not been there before, observing *Figure* 4.2. These were also four of the areas previously identified as regions of higher concentration of infected trees, listed in the reports.
Comparing the infected area from year 2016 to the one from 2015, we ca see that there has been a noticeable increase in the number of damaged trees, increasing little more than twofold.

Attack	Date (*yyyy-mm-dd*)	Area (*percentage*)	Area (*hectares*)
Pre-attack	2015-07-11	4.97	213.2
Post-attack	2016-08-24	11.11	476.7

Comparing the results calculated for the area in 2016, shown above, with the damaged area recorded by the forest rangers, which accounted for 366.6 *hectares*, we can deduce the algorithm identified damaged regions with an approximate error of 30%. This is a significant error, however, it is important to note that four major infected regions have been identified. Given the fact that the resolution of the images is only 10 meters, meaning individual trees cannot be spotted, this proves to be a significant result.

Chapter 5

Conclusions

As this project has now come to an end, it can be concluded that the main objective has been accomplished. An algorithmic solution capable of recognising bark beetle infestations through analysing satellite data has been developed. Since it uses publicly available satellite data provided by the *Sentinel-2* satellite programme, it can practically analyse data for any region. Moreover, its implementation using Python allows for it to be easily deployed and used by local communities.
Of course this is far from an ideal solution, capable of always spotting unhealthy regions of forest. However, taking into account that previously there only existed theoretical solutions, which due to their complexity were not implemented into real-life usage, this algorithmic solution can at least help local forest rangers to keep track of the development in their forests.
This project also contributed to the community by openly releasing two new data-sets. The first contains information about 24 areas that have been impacted by the bark beetle, while the second contains detailed information of exactly how much land had been impacted by the bark beetle attack. Both data-sets can be used for future development and testing of new and existing detection algorithms.

5.1 System Evaluation

Even though the quality of the data obtained and later used for testing, had been questioned since there were many limitations associated with it, the algorithm achieved 85% success rate in identifying regions that had been impacted by the bark beetle infestations, and 87.5% success rate in recognising change in the forest area, be that negative or positive. This is higher than initially expected, especially taking into account the resolution of just 10 meters, which means an individual tree cannot be spotted. What is more, the regions used in this project had been very diverse, ranging from mountainous regions slightly covered with snow, to planes densely populated with forests. However, the diversity of the data was only a coincidence and there were far too little samples of each type of region used to draw any strong conclusions relating to the performance of the algorithm under different conditions. Obtaining larger data-sets, where these diverse regions would be more fairly represented, would be a very good continuation of testing

and further development of the algorithm.
Moreover, there was only one region to which the algorithm's performance could be more strictly compared in terms of recognition of land area impacted by the attack. Since the research team lead by *Professor Kranjcic* did not publish results of their model for the Milanov Vrh, this is the first time the data-set is being used. The algorithm over-estimated the impacted area by approximately 30%, which is quite a lot, however, considering the fact that this is a first of such algorithms, this can be a very good starting point. What is more, the algorithm estimated more area than it was recorded by the rangers to be damaged, which is in terms of prevention of potential bark beetle attacks a desirable action. This shows the potential of the algorithm to be used as an assistant tool for the forest rangers to use to help them monitor the health of their forests.
Through the development of the model, a cloud masking component has been customised and fine-tuned, which achieved an *f-score* of 90.6%, showing that the model can also analyse real-life data even with impurities such as clouds. This only shows its potential for becoming a real assistant tool for the forest rangers.

Besides the algorithmic solution, also two new data-sets were created and openly released. The first data-set contains information about 24 areas that had been impacted by the bark beetle attack, spread across eight regions in Slovenia and Croatia. For each region two satellite images have been obtained and pre-processed, one prior and one after the attack. While it is unclear to what extent the regions were impacted by the bark beetle attack, this is the first openly avialable collection that can be accessed online. The second data-set contains a detailed information of how much land had been impacted by the bark beetle attack in the region of Milanov Vrh. This can serve as a performance measure for future detection algorithms.

5.2 What I Learned

This project has been an incredible and a truly rewarding journey. I have had the opportunity to develop various skills as well as familiarise myself with the new areas that always interested me greatly. From observing a real-life problem, to coming up with an idea for a potential solutions, and to bringing the idea to life, has had its fair share of difficult problems.
While I had the idea of how I wanted to tackle the problem, I definitely lacked in knowledge of the area into which I indulged. Familiarising myself with new areas had been a challenge, especially because with such vast amounts of new knowledge, I easily got overwhelmed and did not know exactly where to search for solutions and answers, even more, asking the right questions was at times a challenge on its own. However, through constant support and encouragement from my mentor, I managed to grasp the understanding of newly discovered topics to the extent I could produce a product that achieved my aims.

The biggest problem I was tasked with through this journey was searching for publicly available data. Since bark beetles are such a big threat and since much of the resources are dedicated to tackling this problem, I was expecting a lot of data to be available in terms of the regions that had been impacted and the amount of land that had been damaged. However, the reality turned out to be different. This forced a lot of brainstorming

for modifying the initial plan and idea, from taking a machine learning approach to developing a change-based detection algorithm. Next challenging task was familiarising myself with the satellite data. In the first couple of weeks it had been a great struggle obtaining even a simple picture. Having very little experience with JavaScript and not being familiar with the satellites and their data, made the beginnings rough. However, after learning the basics of the new programming language, and reading through the satellite documentations, I developed enough skills to start playing around with different data-sets provided by various satellite networks. The next step was to develop enough understanding of vegetation and forestry, to further develop the idea of how the colour change in the images can be evaluated. In contrast to my expectations, observing the change with a naked eye was not as easy. This was done by reading various different scientific articles, which on its own have been a new concept to me.

To conclude, working on this project has thought me a great deal. Not only in terms of newly acquired technical skills, but also about myself. On one hand, it has showed me that I can navigate through hard problems which I would previously avoid, while it simultaneously thought me to re-asses my capabilities and not over-estimate myself. Many times a task that was initially classified as a simpler one, turned out to be a big challenge, requiring much more time than initially anticipated.

As this project has now come to an end, I must admit that the algorithm performed much better than I initially anticipated and I hope that it will be able to help local communities in monitoring their forests.

5.3 What the Future Holds

While a solution that could potentially help local communities fight against bark beetles, has been developed, the lack of available data prevented from fully estimating its capabilities and identifying the associated limitations. It would be very much exciting if more data could be obtained, which could be used for upgrading the algorithm in becoming more efficient and more thoroughly testing its capabilities. Partnering with different organisations responsible for monitoring health of forest could be a great start of obtaining more data. This introduces another interesting opportunity of attempting to deploy the algorithm into the real-world, and observing whether it can be of help to the local communities and organisations. This would also provide us with the potential improvements, needed for the algorithm to be suitable for deployment and usage in the real World.

Bibliography

[1] European Space Agency.

[2] European Space Agency. Sen2cor configuration and user manual. Technical Report S2-PDGS-MPC-L2A-SUM-V2.10, European Space Agency, 8-10 rue Mario Nikis, Paris, France, 2021.

[3] European Space Agency. Copernicus programme - Introduction. https://www.copernicus.eu/en/about-copernicus, 2022. [Online; accessed 09-March-2022].

[4] European Space Agency. Sentinel-2 - Introduction. https://sentinels.copernicus.eu/web/sentinel/missions/sentinel-2, 2022. [Online; accessed 09-March-2022].

[5] European Space Agency. Sentinel-2 - Mission. https://www.esa.int/Applications/Observing_the_Earth/Copernicus/The_Sentinel_missions, 2022. [Online; accessed 09-March-2022].

[6] European Space Agency. Sentinel-2 - Mission Description. https://sentinel.esa.int/web/sentinel/missions/sentinel-2, 2022. [Online; accessed 09-March-2022].

[7] Matej Batič. https://medium.com/sentinel-hub/sentinel-hub-cloud-detector-s2cloudless-a67d263d3025. [Online; accessed 09-March-2022].

[8] Mario Bretfeld, Heather N. Speckman, Daniel P. Beverly, and Brent E. Ewers. Bayesian predictions of bark beetle attack and mortality of three conifer species during epidemic and endemic population stages. *Frontiers in Forests and Global Change*, 4, 2021.

[9] Vojtěch Bárta, Jan Hanuš, Lumír Dobrovolný, and Lucie Homolová. Comparison of field survey and remote sensing techniques for detection of bark beetle-infested trees. *Forest Ecology and Management*, 506:119984, 2022.

[10] Vojtěch Bárta, Petr Lukeš, and Lucie Homolová. Early detection of bark beetle infestation in norway spruce forests of central europe using sentinel-2. *International Journal of Applied Earth Observation and Geoinformation*, 100:102335, 2021.

[11] Yale Climate Connection. Global warming fuels bark beetles, tree-killing menaces. https://storymaps.arcgis.com/collections/

a220e2d0609c45d6976e74cff7a54bf3?item=1, 2022. [Online; accessed 09-April-2022].

[12] Placer Resource Conservation District. Tree Mortality. https://placerrcd.org/landowner-assistance/forestry/tree-mortality/, 2022. [Online; accessed 09-March-2022].

[13] Google Earth Engine. Google Earth Engine - Introduction. https://earthengine.google.com/, 2022. [Online; accessed 09-March-2022].

[14] Ranga B. Myneni et al. The Interpretation of Spectral Vegetation Indexes. https://d1wqtxts1xzle7.cloudfront.net/41885663/'The_interpretation_of_spectral_vegetatio20160202-12274-9izmz4-with-cover-page-v2.pdf', 2022. [Online; accessed 09-March-2022].

[15] Hugh Evans. The threat to uk conifer forests posed by ips bark beetles. Technical Report ISBN: 978-1-83915-008-1, Forest Research, 231 Corstorphine Road, Edinburgh, United Kingdom, 2021.

[16] Angel Fernandez-Carrillo, Zdeněk Patočka, Lumír Dobrovolný, Antonio Franco-Nieto, and Beatriz Revilla-Romero. Monitoring bark beetle forest damage in central europe. a remote sensing approach validated with field data. *Remote Sensing*, 12(21), 2020.

[17] Christopher J. Fettig, Kier D. Klepzig, Ronald F. Billings, A. Steven Munson, T. Evan Nebeker, Jose F. Negrón, and John T. Nowak. The effectiveness of vegetation management practices for prevention and control of bark beetle infestations in coniferous forests of the western and southern united states. *Forest Ecology and Management*, 238(1):24–53, 2007.

[18] Christopher J. Fettig and Stephen R. McKelvey. Resiliency of an interior ponderosa pine forest to bark beetle infestations following fuel-reduction and forest-restoration treatments. *Forests*, 5(1):153–176, 2014.

[19] Forestry.com. Is the bark beetle still here? https://www.forestry.com/editorial/is-the-bark-beetle-still-here/, 2021. [Online; accessed 09-March-2022].

[20] Demian F. Gomez, Haley M.W. Ritger, Christopher Pearce, Jeffrey Eickwort, and Jiri Hulcr. Ability of remote sensing systems to detect bark beetle spots in the southeastern us. *Forests*, 11(11), 2020.

[21] Nikola Kranjčić and Milan Rezo. Satellite-based hyperspectral imaging and cartographic visualization of bark beetle forest damage for the city of Čabar. *Tehnički glasnik*, 12:39–43, 03 2018.

[22] Sinergise Matej Aleksandrov. Module for making pixel-based classification on Sentinel-2 L1C imagery. https://github.com/sentinel-hub/sentinel2-cloud-detector/blob/master/s2cloudless/cloud_detector.py, 2021. [Online; accessed 09-April-2022].

[23] Land Trust of North Alabama. Layers of a Tree. https://www.landtrustnal.org/2018/10/26/timber-management-on-chapman-mountain/layers-of-a-tree/, 2022. [Online; accessed 09-March-2022].

[24] Forest Service of the Republic of Slovenia. Interactive Map. https://prostor.zgs.gov.si/pregledovalnik/, 2021.

[25] University of Wisconsin-Milwaukee. European Elm Bark Beetle (Family Curculionidae). https://insider.si.edu/2010/03/new-acquisition, 2014. [Online; accessed 09-March-2022].

[26] Pinterest. Beetles Mariachi Band. https://www.pinterest.co.uk/pin/239183430195906614/, 2022. [Online; accessed 09-March-2022].

[27] Anastasiia Safonova, Siham Tabik, Domingo Alcaraz-Segura, Alexey Rubtsov, Yuriy Maglinets, and Francisco Herrera. Detection of fir trees (abies sibirica) damaged by the bark beetle in unmanned aerial vehicle images with deep learning. *Remote Sensing*, 11(6), 2019.

[28] European Space Agency Sentinel satellites. Europe's eyes on Earth. https://www.copernicus.eu/en, 2021. [Online; accessed 09-March-2022].

[29] Colorado Forest Service. Forest Health Highlights: Colorado. https://storymaps.arcgis.com/collections/a220e2d0609c45d6976e74cff7a54bf3?item=1, 2022. [Online; accessed 09-March-2022].

[30] Mundi Web Services. Advanced Geo-Informational Systems. https://www.sinergise.com/, 2021. [Online; accessed 09-March-2022].

[31] Mundi Web Services. Grow Your Business from Space. https://mundiwebservices.com/, 2021. [Online; accessed 09-March-2022].

[32] Sinergise. SentinelHub API - Introduction. https://sentinelhub-py.readthedocs.io/en/latest/examples/process_request.html, 2022. [Online; accessed 09-March-2022].

[33] Smithsonia. Eighty-thousand bark beetles enter National Museum of Natural History collections. https://insider.si.edu/2010/03/new-acquisition, 2022. [Online; accessed 09-March-2022].

[34] United States Geological Survey. Remote Sensing - Definition. https://www.usgs.gov/faqs/what-remote-sensing-and-what-it-used#, 2022. [Online; accessed 09-March-2022].

[35] Earth Observing System. NDVI: Normalised Difference Vegetation Index. https://eos.com/make-an-analysis/ndvi/, 2021. [Online; accessed 09-March-2022].

[36] Katelyn Tarrio, Xiaojing Tang, Jeffrey G. Masek, Martin Claverie, Junchang Ju, Shi Qiu, Zhe Zhu, and Curtis E. Woodcock. Comparison of cloud detection algorithms for sentinel-2 imagery. *Science of Remote Sensing*, 2:100010, 2020.

[37] Klokan Technologies. https://epsg.io/4326. [Online; accessed 09-March-2022].

[38] Wikipedia. Sentinel mission. https://en.wikipedia.org/wiki/Sentinel-2, 2021. [Online; accessed 09-March-2022].

[39] Wikipedia. Bark beetle — Wikipedia, the free encyclopedia. http://en.wikipedia.org/w/index.php?title=Bark%20beetle&oldid=1074116004, 2022. [Online; accessed 10-April-2022].

[40] Wikipedia. Cirrus cloud — Wikipedia, the free encyclopedia. http://en.wikipedia.org/w/index.php?title=Cirrus%20cloud&oldid=1079581592, 2022. [Online; accessed 09-April-2022].

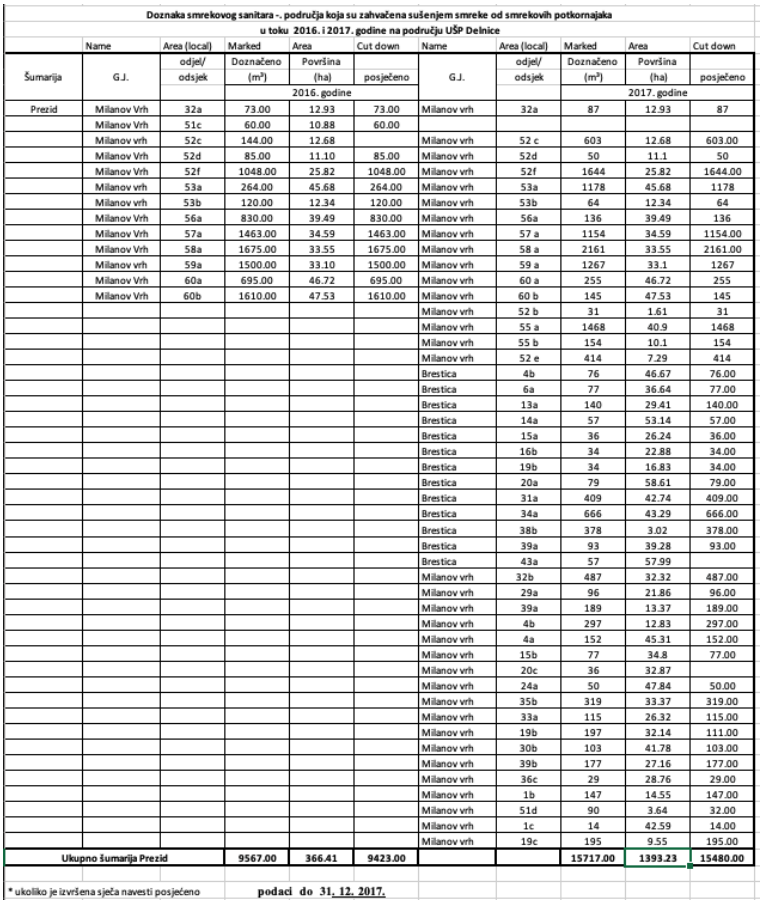
[41] Michael A Wulder and Steven E Franklin. *Remote sensing of forest environments: concepts and case studies*. Springer Science & Business Media, 2012.

[42] Qingfu Xiao and E. Mcpherson. Tree health mapping with multispectral remote sensing data at uc davis, california. *Urban Ecosystems*, 8:349–361, 12 2005.

Appendix A

Raw Data for Milanov Vrh

Since the data was obtained in the format of a large excel file, I decided to provide a screenshot of the data. The full file can be accessed on my github repository available through this link: `https://github.com/Pompey21/Don-t-Let-the-Bark-Beetle-Bite-You`.



Doznaka smrekovog sanitara -. područja koja su zahvačena sušenjem smreke od smrekovih potkornajaka
u toku 2016. i 2017. godine na području UŠP Delnice

	Name	Area (local)	Marked	Area	Cut down	Name	Area (local)	Marked	Area	Cut down
Šumarija	G.J.	odjel/ odsjek	Doznačeno (m^3)	Površina (ha)	posječeno	G.J.	odjel/ odsjek	Doznačeno (m^3)	Površina (ha)	posječeno
			2016. godine					2017. godine		
Prezid	Milanov Vrh	32a	73.00	12.93	73.00	Milanov vrh	32a	87	12.93	87
	Milanov Vrh	51c	60.00	10.88	60.00					
	Milanov vrh	52c	144.00	12.68		Milanov vrh	52 c	603	12.68	603.00
	Milanov Vrh	52d	85.00	11.10	85.00	Milanov vrh	52d	50	11.1	50
	Milanov Vrh	52f	1048.00	25.82	1048.00	Milanov vrh	52f	1644	25.82	1644.00
	Milanov vrh	53a	264.00	45.68	264.00	Milanov vrh	53a	1178	45.68	1178
	Milanov vrh	53b	120.00	12.34	120.00	Milanov vrh	53b	64	12.34	64
	Milanov Vrh	56a	830.00	39.49	830.00	Milanov vrh	56a	136	39.49	136
	Milanov Vrh	57a	1463.00	34.59	1463.00	Milanov vrh	57 a	1154	34.59	1154.00
	Milanov Vrh	58a	1675.00	33.55	1675.00	Milanov vrh	58 a	2161	33.55	2161.00
	Milanov vrh	59a	1500.00	33.10	1500.00	Milanov vrh	59 a	1267	33.1	1267
	Milanov Vrh	60a	695.00	46.72	695.00	Milanov vrh	60 a	255	46.72	255
	Milanov Vrh	60b	1610.00	47.53	1610.00	Milanov vrh	60 b	145	47.53	145
						Milanov vrh	52 b	31	1.61	31
						Milanov vrh	55 a	1468	40.9	1468
						Milanov vrh	55 b	154	10.1	154
						Milanov vrh	52 e	414	7.29	414
						Brestica	4b	76	46.67	76.00
						Brestica	6a	77	36.64	77.00
						Brestica	13a	140	29.41	140.00
						Brestica	14a	57	53.14	57.00
						Brestica	15a	36	26.24	36.00
						Brestica	16b	34	22.88	34.00
						Brestica	19b	34	16.83	34.00
						Brestica	20a	79	58.61	79.00
						Brestica	31a	409	42.74	409.00
						Brestica	34a	666	43.29	666.00
						Brestica	38b	378	3.02	378.00
						Brestica	39a	93	39.28	93.00
						Brestica	43a	57	57.99	
						Milanov vrh	32b	487	32.32	487.00
						Milanov vrh	29a	96	21.86	96.00
						Milanov vrh	39a	189	13.37	189.00
						Milanov vrh	4b	297	12.83	297.00
						Milanov vrh	4a	152	45.31	152.00
						Milanov vrh	15b	77	34.8	77.00
						Milanov vrh	20c	36	32.87	
						Milanov vrh	24a	50	47.84	50.00
						Milanov vrh	35b	319	33.37	319.00
						Milanov vrh	33a	115	26.32	115.00
						Milanov vrh	19b	197	32.14	111.00
						Milanov vrh	30b	103	41.78	103.00
						Milanov vrh	39b	177	27.16	177.00
						Milanov vrh	36c	29	28.76	29.00
						Milanov vrh	1b	147	14.55	147.00
						Milanov vrh	51d	90	3.64	32.00
						Milanov vrh	1c	14	42.59	14.00
						Milanov vrh	19c	195	9.55	195.00
Ukupno šumarija Prezid			**9567.00**	**366.41**	**9423.00**			**15717.00**	**1393.23**	**15480.00**

* ukoliko je izvršena sječa navesti posječeno

podaci do 31. 12. 2017.

Figure A.1: Screenshot of the raw data of Milanov Vrh area.

Appendix B

Results

B.1 Cloud Masking

The full results from the *Cloud Masking Algorithm* section are available in the *Table* B.1. There were in total 42 different combinations of parameters tested, however, only the ones that produced valid results and did not return an error had been displayed in the table, summing up to 33 all-together.

B.2 Damaged Areas

The full results from the *Damaged Areas Testing* section are available in the *Figure* B.1. There were in total 24 different areas the algorithm's performance was tested on. With green colour are indicated those regions for which the algorithm recognised an increase in damaged area, with orange those for which the algorithm recognised a decrease in damaged area, and with red with red those that have not been detected.

of parameters tested, however, only the ones that produced valid results and did not return an error had been displayed in the table, summing up to 33 all-together.

Number	Threshold	average	dilation	all bands	precision	recall	f-measure
1	0.6	1	1	FALSE	0.873107682	0.9423	0.9064
2	0.5	4	4	FALSE	0.9347	0.8114	0.8687
3	0.6	8	5	FALSE	0.9755	0.7089	0.8211
4	0.7	6	5	FALSE	0.9877	0.6120	0.7557
6	0.5	2	2	FALSE	0.9917	0.5578	0.7140
7	0.7	7	6	FALSE	0.9917	0.5578	0.7140
8	1	2	4	FALSE	0.9646	0.6942	0.8073
9	0.9	1	8	FALSE	0.9277	0.8393	0.8813
10	0.7	7	8	FALSE	0.9570	0.7111	0.8159
11	0.9	2	7	FALSE	0.9637	0.7205	0.8245
12	0.4	3	2	FALSE	0.8361	0.9602	0.8939
13	0.5	6	6	FALSE	0.8900	0.8857	0.8878
14	0.4	7	6	FALSE	0.9787	0.6174	0.7571
15	0.2	5	6	FALSE	0.9011	0.8778	0.8893
16	0.8	7	4	FALSE	0.8499	0.9548	0.8993
17	0.4	1	5	TRUE	0.9782	0.6767	0.8000
18	0.6	1	1	TRUE	0.9637	0.6979	0.8095
19	0.5	4	4	TRUE	0.9704	0.6800	0.7997
20	0.6	8	5	TRUE	0.9322	0.8230	0.8742
21	0.7	6	5	TRUE	0.8763	0.9058	0.8908
22	1	7	3	TRUE	0.9243	0.8228	0.8706
23	0.5	2	2	TRUE	0.9113	0.8464	0.8779
24	1	2	4	TRUE	0.9913	0.5854	0.7361
25	0.9	1	8	TRUE	0.9243	0.8228	0.8706
26	0.7	7	8	TRUE	0.9787	0.6174	0.7571
27	0.9	2	7	TRUE	0.9601	0.7487	0.8413
28	0.4	3	2	TRUE	0.8900	0.9068	0.8983
29	0.5	6	6	TRUE	0.9737	0.6326	0.7670
30	0.4	7	6	TRUE	0.9646	0.6942	0.8073
31	0.1	7	8	TRUE	0.9871	0.5635	0.7175
32	0.1	4	5	TRUE	0.9347	0.8114	0.8687
33	0.2	4	2	FALSE	0.8731	0.9423	0.9064

Table B.1: Results of the cloud masking algorithm for the chosen set of parameter combinations.

	Region	Year (pre)	percent	Year (post)	percent		Pass
1	Stari Trg 1	2016	1.732	2019	18.0498		
2	Stari Trg 2	2016	1.057	2019	8.073		
3	Krka - Zuzemberk 1	2016	1.987	2018	7.0779		
4	Krka - Zuzemberk 2	2016	2.212	2018	6.337		
5	Krka - Zuzemberk 3	2016	2.23	2018	5.2317		
6	Bled 1	2016	15.19	2018	15.887		
7	Bled 2	2015	15.973	2018	13.555		
8	Bled 3	2016	9.051	2018	13.542		
9	Bled 4	2016	4.861	2018	10.618		
10	Bled 5	2016	28.506	2018	27.45		
11	Bled 6	2016	29.915	2018	41.251		
12	kanomlja-idrija 1	2015	5.555	2018	11.377		
13	kanomlja-idrija 2	2015	6.256	2018	17.94		
14	kanomlja-idrija 3	2015	15.748	2020	3.856		
15	skocjan 1	2017	9.0873	2018	14.3412		
16	skocjan 2	2016	10.21	2018	18.7143		
17	skocjan 3	2016	3.319	2017	1.7912		
18	skocjan 4	2017	3.012	2018	6.605		
19	skocjan 5	2016	1.24	2020	1.49		
20	skocjan 6	2015	8.138	2020	0.8448		
21	Osojnica 1	2016	9.11	2017	22.159		
22	Osojnica 2	2016	14.923	2017	30.834		
23	Cabar 1	2015	1.404	2018	49.208		
24	Cabar 2	2015	7.351	2016	5.465		
	zled						
	podlubniki						
	cabar						

Figure B.1: Screenshot of algorithm's performance on recognising each individual region.