GIMA
Geographical Information Management and Applications

Deep Learning Method for forest fire detection and simulation of wildfire expansion using sentinel-2 images.

Final thesis report

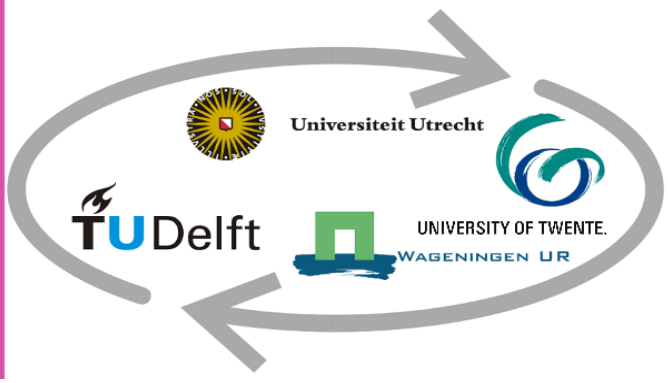
Name (student ID): Karalidis Konstantinos (1631780)

Date: 28/02/2023

Responsible Professor: Dr.ir. Peter van Oosterom

Supervisor: Dr. Azarakhsh Rafiee

Place of research: TU Delft



Final Thesis Report

Deep Learning Method for forest fire detection and simulation of wild-fire expansion using sentinel-2 images.

By

Karalidis Konstantinos

k.karalidis@students.uu.nl
UU student number: 1631780

In partial fulfillment of the requirements for the degree of
Master of Science (MSc)
Geographical Information Management and Applications (GIMA)

Responsible Professor: Dr.ir. Peter van Oosterom
Supervisor: Dr. Azarakhsh Rafiee
Place of research: TU Delft

To my family

Acknowledgments

This thesis could not have been realized without some people's contributions, so I feel obliged to express my thanks to them.

First of all, I would like to thank the supervisor Professor Azarakhsh Rafiee for her impeccable guidance and her catalytic role in processing my thesis. Moreover, I would like to thank my responsible professor Peter van Oosterom for his useful comments and guidance throughout the thesis process.

Finally, all of this could not have been realized without my parents who with their support all these years they have given me the possibility to develop so much both academically and personally. I thank them from the bottom of my heart, and I promise them that I will always try to do them proud.

Athens, 27 February 2023

Abstract

One of the main threats that the world faces right now is natural disasters, which every year cause irreparable damage to both the human and natural environments. Forest fires have emerged as one of the most important threats that humanity faces in the 21st century. On the other hand, Convolutional neural networks as a part of machine learning have gained a lot of attention over the last years due to the rise of artificial neural networks. Convolutional neural networks are distinguished by their high accuracy and ability to recognize difficult image patterns. In addition to that, cellular automata have been widely used for forest fire spread simulation as it can easily adapted to dynamic environments with spatiotemporal development. This thesis aims to integrate the emerging evolution of artificial intelligence with cellular automata in order to detect forest fire outbreaks and simulate forest fire spreads by using sentinel-2 images. The results of this study show that the inception V3 model, as part of convolutional neural networks, can detect very well fire outbreaks with low data availability and that cellular automata can simulate the spread of the forest fire very accurately.

Key words: Convolutional neural networks, inception V3, Cellular automata, satellite images, wildfire modelling, Fire detection

Tables

Table 1: Relatively studies on CNN 15
Table 2: Sources of data 23
Table 3:Wind shear for various terrain types (Engineeringtoolbox.com, 2020) 27
Table 4: Land cover assumptions 30
Table 5:Values of $Pvegetation$, $Pdensity$ probabilities and constant parameters 31
Table 6: Fire spread conditions. 31
Table 7: Confusion Matrix results 45
Table 8: Confusion Matrix results of the simulation model 45

Figures

Figure 1: Comparison of the avg. number of wildfires from 2006 to 2021 with the number of fires in the current year 9
Figure 2: convolutional layer (Source: O'Shea & Nash, 2015) 14
Figure 3: CNN model (Source: Dong et al., 2020) 14
Figure 4: On the left a 2-D cellular automata neighborhoods (Source: Crooks, 2017) and on the right the temporal aspect of Cellular automata (Source: Giabbanelli, 2019) 17
Figure 5: Steps of fire detection and fire spread simulation. 18
Figure 6: A 3x3 convolution is replacing a 5x5 (Source: Szegedy et al., 2016) 19
Figure 7: A 3x3 layer replaced by a 3x1 layer (Source: Szegedy et al., 2016) 19
Figure 8: Inception module without factorizing (Source: Szegedy et al., 2015) 20
Figure 9: Inception V3 module (Source: Szegedy et al., 2015)) 20
Figure 10: The location of Kineta in the Greek geographical space 21
Figure 11: 3D model of Kineta area (Source: Google earth) 22
Figure 12: On the left is a B12, B11, B4 combined image and on the right a true color image 22
Figure 13: Samples of the sentinel-2 images. On top are the fire images and, in the bottom, the non-fire images. 23
Figure 14: Number of cells for each category 24
Figure 15: Land cover classification in Kineta 25
Figure 16: Possible direction of fire expansion (Alexandridis et al., 2008). 25
Figure 17: Cell neighbourhood 26
Figure 18:Average wind speed in Kineta. 28
Figure 19: Slope values in Kineta 29
Figure 20: Aspect values in Kineta 30
Figure 21: Fire detection 32
Figure 22: Detection of Non fire images 34
Figure 23: Confusion Matrix 37
Figure 24: Confusion Matrix of the model 37
Figure 25: Testing on larger images 38
Figure 26: Fire spread simulation each time step 40
Figure 27: Area of fire spread every hour. 44
Figure 28:Actual burned area in Kineta 44

Figure 29: Actual burned and simulated fire area. 46
Figure 30: Map display using geemap package. 56
Figure 31: Image collection and visualization 56
Figure 32: Extraction of images 57
Figure 33: ImageDataGenerator 58
Figure 34: Inception V3 model structure 58
Figure 35: Freezing the layers 59
Figure 36: Validation and training accuracy after and before freezing the model. On the left is before and on the right is after. 59
Figure 37: NetLogo interface 60
Figure 38: Setup procedure 60
Figure 39: Calculation of fire percentage 61
Figure 40: Counting of total neighbors. 62
Figure 41: Fire birth funtion 62
Figure 42: Fire spread code 63

Contents

Abstract **5**

Tables **6**

Figures **6**

1.Introduction **9**

1.1 Context and problem statement 9

1.2 Research Objectives 11

1.3 Research limitations and scope 11

1.4 Research structure 12

2. Theoretical framework **13**

2.1 Convolutional Neural Networks 13

2.1 Cellular automata 16

3. Methodology **18**

3.1 Inception-v3 19

3.2 Study area and data collection 21

3.2.1 Study area 21

3.2.2 Data collection 22

3.3 Land cover classification 24

3.4 Fire spread model 25

3.4.1 The effect of wind 26

3.4.2.The effect of terrain 28

3.5. Transition rules 30

4. Results **32**

4.1 Forest fire prediction 32

4.2 Forest fire spread simulation 40

4.3 Validation test 44

5. Conclusion **47**

6. Discussion **48**

Bibliography **50**

Appendices **55**

Appendix I – Land cover reclassification 55

Appendix II: Steps for image collection 56

Appendix III: Steps of fire detection 58

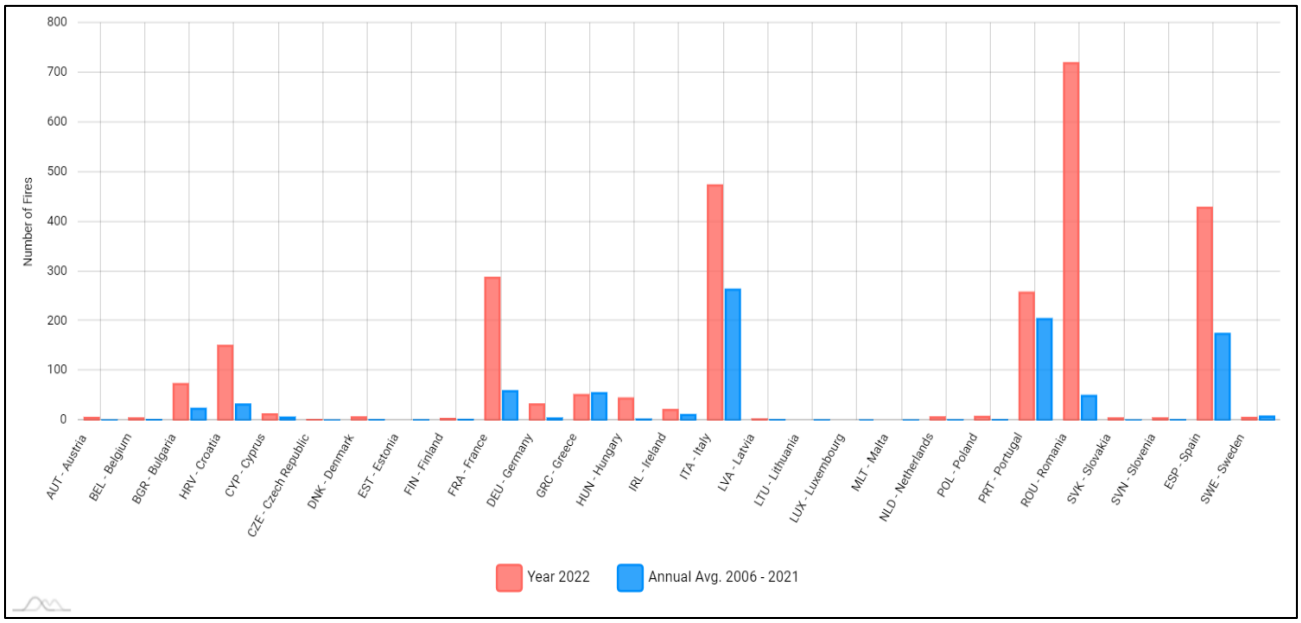
Appendix IV: Description of simulation model 60

1. Introduction

1.1 Context and problem statement

It is well known that forest is considered one of the biggest terrestrial ecosystems by playing a significant role in the overall ecology. They are particularly successful in preserving the quality of the soil and water as well as regulating the climate. On the other hand, forest fires are an important component in disturbing this natural equilibrium (Z. Wu et al., 2022). Over the last years, there has been a significant increase in the number of forest fires which caused serious environmental disasters and endangered both human and animal life (Mazzeo et al., 2022). The figure below shows the annual average number of fires in several European countries from 2006 to 2021 as well as the number of fires since the beginning of that year.

Figure 1: Comparison of the avg. number of wildfires from 2006 to 2021 with the number of fires in the current year



Data: EFFIS Statistical Portal

According to the figure above, it can be seen that many countries face a large number of fire outbreaks each year, reaching in 2022 to exceed the average annual number of fire outbreaks since 2006. Based on all that has already been formulated, it can be understood that forest fires are an important problem that cannot accommodate any other delays in dealing with it.

Over the last years, due to the continuous increase of fire outbreaks international space agencies like NASA see the urgency to better utilize satellite observations to detect fires, particularly through sensors like SEVIRI (Spinning Enhanced Visible and InfraRed Imager), AVHRR (Advanced Very High-Resolution Radiometer), MODIS (Moderate Resolution Imaging Spectroradiometer), and VIIRS (Visible Infrared Imaging Radiometer Suitability (Mazzeo et al., 2022). These sensors can provide almost every 1- or 2-days data of forest fire outbreaks with relatively low spatial resolution. According to Hu et al. (2021) Earth observation satellites have been used to regularly track fire activity across large areas in two different approaches: 1) to locate the places of actively burning areas, and 2) to map the geographical extent of the burnt traces. Moreover, as satellite remote sensing technology advances, more and more researchers are choosing to employ satellite multispectral images to detect

forest wildfires (Hong et al., 2022). In addition to that, CNNs, (Convolutional Neural Networks,) have made enormous breakthroughs in machine learning, which have stimulated the development of innovative designs and approaches for a variety of applications, including active fire identification in satellite images (de Almeida Pereira et al., 2021). Moreover, the Inception V3 models have gained a lot of attention the last years due to their high accuracy in image recognition.

Recent research has become interested in network structure based on Inception v3 integrated with transfer learning because of its outstanding performance on a variety of small datasets (Dong et al., 2020). According to C. Wang et al (2019) transfer learning emerged as a reduction method of time and computer resources due to the high rise in the neural network's performance and depth. The basic concept of transfer learning is to facilitate the training of a new model by transferring the trained model parameters to the new model. This exchange of parameters allows the new model to not begin from scratch as the majority of the networks do, as well as the learning efficiency of the new model may be expedited and optimized given that some data or tasks are important (C. Wang et al., 2019). Szegedy et al (2016) introduced the Inception model as a deep CNN architecture, in the Large-Scale ImageNet Visual Identification Challenge 2014 with the intention of lessening the impact of processing efficiency and low parameters in application scenarios (Cao et al., 2021).

The detection of the fire in early stages is the first step of dealing with fire outbreaks, the next step is to predict and simulate the fire expansion. Wu et al. (2022) mentioned that the prediction of the wildfire expansion is based mainly on mathematical models in recent years. Moreover, they believe that fire spread models could be separated into three categories, i) the physical model which is based on the law of conservation of energy ii) the empirical models which analyze the historical data, and they combine them with current meteorological factor and iii) the semiempirical forest fire spread models which is a combination of the two previous models (Wu et al., 2022). Nowadays, artificial neural networks (ANNs) have been effectively applied to forecast different kinds of issues by using information gathered through self-learning techniques (Safi & Bouroumi, 2013). This method has very similarities with the second category of the fire spread models.

Until now there are several models that have been applied to simulate forest fire spreading. Byari et al. (2022) applied a multi-scale 3D cellular automata model for forest fire spreading by taking into consideration the diversity of the environment based on several transition rules. Bodrožić et al. (2006) used CA to predict forest fire spread. They acquired data from the Corine, and they assign two cell values burned or no burned. After that they set some transition rule and simulate the fire spreading in two situations, with wind and with no wind. Moreover, Almeida and MacAu (2011) applied a stochastic cellular automata model in a flat ground with no wind. They used this model in order to represent behaviors of the vegetation combustion and igniting process during the fire spreading. Another very interesting study if this of Alexandridis et al. (2008). They used CA to simulate and analyze how a forest wildfire spreads in a mountainous environment. They include elements such as vegetation variety and density as well as the wind's speed and direction. Furthermore, they separate the cells state into four categories. The first state the cell has no forest fuel, this is because they wanted to represent areas like city, and barren lands that cannot been burned. The second state is the cell that has forest fuel but is not yet ignited. The third state the cell has already been ignited and finally the last state is the cell that has already been ignited and burned. In addition, Wang et al (2017) to examine the influence of several factors in the spreading of forest wildfire. They separated it into three factors. The Dynamic Factor-Meteorology which analyzes the meteorological value change such as the speed of the wind and the temperature. The Static Factor-Terrain analyzes factors that are related to the Ter-

rain such as slope, and the last factor is the Combustibles which are refereeing to the flammable materials of the forest environment.

According to the above, it can be well understood that CA has been applied many times in forest fire spread simulation. There are many factors that influence the course of the fire such as wind speed, the topography of the terrain and the vegetation variety. Moreover, the criteria for the transition rules play important role in the model. The criteria focus on the likelihood of a cell to be ignited or not. The cell burns or remains unburned depending on whether the conditions for ignition are met at each point in the time loop. This study scopes to take into consideration all the above factors and make the model as realistic as possible.

Taking everything into consideration, this research works on a CNN model for wildfire outbreaks detection using sentinel-2 images. Moreover, the study is extended to simulate the course of a fire in a forestry area using cellular automata. Finally, the main outcome of this research is to contribute to the fight against forest fires and to constitutes an important tool for dealing with environmental disasters.

1.2 Research Objectives

In the previous section the main problem as well as the context of the research was discussed. In this section the main research objective and the research questions are formulated.

The main objective of this research is to detect fire outbreaks by working with convolutional neural network and simulate fire spread using sentinel-2 images.

This objective has several research questions and sub-questions:

I. To what extent can convolutional neural network detect forest fires?
    - How accurate this model can detect fire outbreaks?
    - How does the model perform on larger images?

I. To what extent can cellular automata simulate forest fire spread?
    - what factors affect the route of a wildfire in a forest area?
    - How accurate can this model simulate the spread of a wildfire by comparing with already fire outbreaks?

1.3 Research limitations and scope

This research will only focus on sentinel-2 images from the period of 2018 until 2022. However, a further analysis to other satellites can be an important topic for future research. The data of the non-fire images that are needed to train the model are filtered to have less than five percent of clouds. This is because the images had to depict the terrain as clearly as possible. Furthermore, the sentinel-2 images are extracted with the band combination B12, B11 and B4 because the fire is more clearly depicted in that band combination. This is explained more thoroughly in the Methodology section.

1.4 Research structure

The present research was structured in such a way, seeking to answer the research questions raised. More specifically:

Chapter 1 presents the problem and context of this research as well as the research questions that lead to the main purpose of the research.

In chapter 2 the theoretical framework is analyzed and explained. More specifically, firstly, the history of the CNN is presented as well as some recent definitions about the model. In addition, in this section the structure of CNN is mentioned and the main characteristics. Furthermore, the recent studies about the application of CNN in fire detection are presented. After that, the history and the definitions of CA are mentioned as well as how the CA is functioning. Finally, this paragraph is also presenting contemporary studies on forest fire spread using CA.

Chapter 3 presents the logic of the methodological framework. First, the inception V3 model is analyzed and explained as well as his structure and characteristics. Moreover, the study area is defined. After that, the source of the data that are collected is presented and the logic behind this specific selection. Furthermore, the method for the classification of images is analyzed and the transition rules for the simulation model are defined.

The chapter 4 presents the implementation of inception V3 model, and the simulation of the fire spread model. The main aim of this section is to answer and evaluate the research questions that was posed in the beginning of this research.

The chapter 5 summarizes the entire thesis and draws the main research conclusions, while the chapter 6 investigates further analysis on forest fire detection and simulation of forest fire spread.

2. Theoretical framework

2.1 Convolutional Neural Networks

Deep learning has recently achieved considerable success in a variety of applications, including image identification, audio recognition, gaming intelligence, natural language processing, and autonomous navigation (Xu & Zhang, 2022). According to Yeturu (2020) Deep learning allows an input to be transformed from one dimensional space to another while Deep networks are used to identify complicated input patterns that are challenging for others to record and maintain them. One of the deep learning models that have attracted a lot of interest because of their outstanding performance in computer vision applications are the Convolutional Neural Networks (Reimers & Requena-Mesa, 2020).

Convolutional Neural Network is a common deep learning architecture that takes inspiration from how living organisms naturally understand their environment. Hubel and Wiesel, 1968) discovered that animal visual cortex cells are in charge of sensing light in receptive fields (Gu et al., 2018). Motivated by this finding, in the 1990s, the neocognitron (Fukushima, 1980) served as the inspiration for the earliest research on contemporary convolutional neural networks by Lecun et al (1998). LeCun et all. (1998) showed that handwritten character recognition can be accomplished using a CNN model that combines simpler characteristics with increasingly more complex ones.

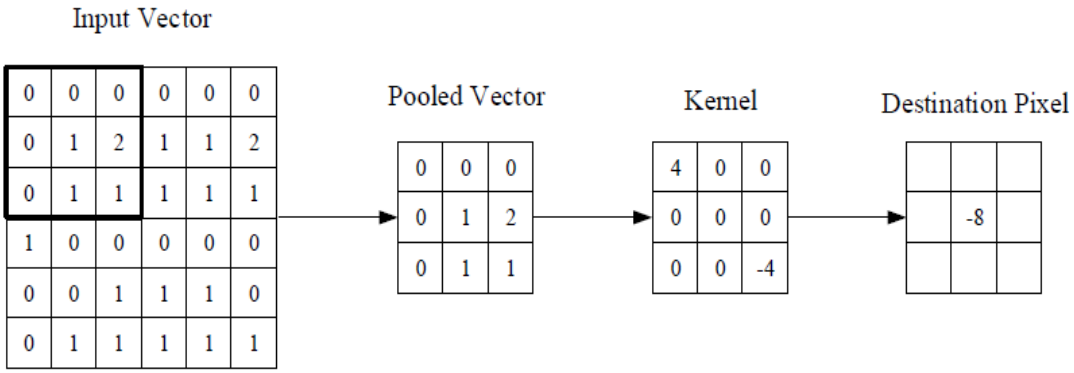
LeCun et all (1998) defined CNN as a neural networks architecture that employ local patterns connections and weight restrictions to include information about the invariances of two-dimensional forms. Today, Duan et al (2022) gives a very accurate definition of the CNN as a typical feedforward neural networks for large-scale feature extraction. In addition to that, Lopez Pinaya et al (2019) mentioned that CNNs are composed of layers of neurons with learnable weights and biases, equivalent to classic deep neural networks (DNNs). Moreover, Xu and Zhang (2022) indicate that the better ability of CNNs to interpret organized arrays of data, such as pictures, talks, and audio signal inputs, sets them apart from other neural networks. Furthermore, it is worth mentioned that techniques based on Convolution Neural Networks demonstrate cutting-edge accuracy on the ImageNet challenge [1] with error rate less than 4% (Jahandad et al., 2019). The primary objective of CNNs is to identify and extract local characteristics from the data. The similarity between specific picture patches and trained kernels is computed in a convolutional layer. The values of nearby pixels are then gathered and blended into one pixel in a pooling layer. As a result, less processing is required, and a feature selection that is resistant to slight changes results from the reduction in data complexity. Then both convolutional and pooling layers are modified. After the first layer of pooling finished, each number of a single pixel represents an entire batch of pixels in the original picture, and after the second layer of pooling is finished, each number represents an entire batch of batches of pixels in the original image (Reimers & Requena-Mesa, 2020).

To understand more deeply O'Shea and Nash (2015) gave a very detailed description of the three components of CNNs. The first component as mentioned above is the convolutional layer which is crucial to CNN operation. The main idea of the convolutional layer is to extract small regions(kernels) of the input data and then recalculates the center pixel of the kernel and replace it by weighted sum of itself and any neighboring pixels, as can be seen in the figure below. This component uses filter sizes to extract the area for example 3x3. Moreover, Strides and padding are two additional crucial ideas in convolutional layers. The

[1] The ImageNet Large Scale Visual Recognition Challenge (ILSVRC) is a project that runs from 2010 where software applications contend to accurately identify objects and scenes.

number of pixels that a kernel or filter slides across the input matrix is referred to as a stride whereas in the padding the filter does not fit the input matrix (Handa et al., 2021).

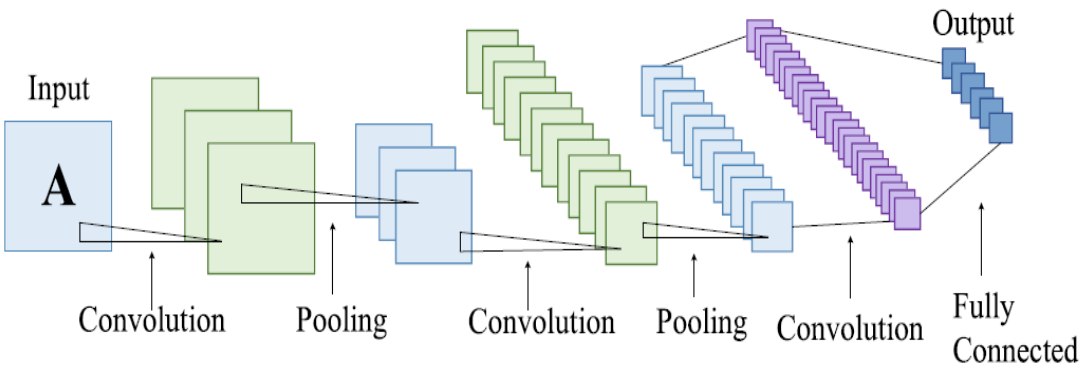
Figure 2: convolutional layer (Source: O'Shea & Nash, 2015)



Another component of the CNNs is the pooling layer. This component scopes to progressively decrease the representation's dimensionality which has as a result to further decrease both the computational complexity of the model and the number of parameters (O'Shea & Nash, 2015). The pooling layer is usually placed in the middle of two convolutional layers. Furthermore, this component of feature maps is linked to their equivalent feature maps from the previous convolutional layer (Gu et al., 2018). According to J. Wu (2017) there are two frequently used variants in the pooling layer the max pooling and the average pooling. The first maps a subrange to its highest value while the second one to its average value (J.Wu, 2017).

The last component is the Fully connected layer. After numerous convolutional and pooling layers, the neurons of these two components are connected to the Fully connected layer. This has as a result to create a global sematic information from the previous layers (Gu et al., 2018). To sum up, in order to produce the output, the CNN first executes a sequence of convolutional and pooling layers before moving on to fully connected layers. The whole process of the Convolutional Neural Networks can be represented in the figure below.

Figure 3: CNN model (Source: Dong et al., 2020)



Over the years convolutional neural networks have been used in many studies in various scientific fields. Rainey et al (2016) used CNN to detect ships on the Singapore harbor by us-

ing satellite images with 0.5 resolution. They extract over 20000 images chips to train the model from WorldView-1 and WorldView-2 satellites. Moreover, a recent study of Hong et al (2022) used CNN to detect active fire in the Guangdong Province, in southern China. They also used satellite images from Himawari-8, but the resolution was between 500-1000 meters. Furthermore, CNNs have been also used in the urban studies. Albert and Gonz (2017) used this model to detect patterns in the land uses in 7 different cities. They obtain satellite images from Google Maps Static API, and they classify into 10 land use types. Convolutional neural networks have also been used in the field of medicine. Charan et al (2018) used this model to classify normal and abnormal breast detection. They used 332 mammograms images of which 189 were normal and the rest were abnormal breasts. According to the above, it is well understood that CNN have an important role in various scientific fields making them an important tool for automatic image recognition.

As far as this research is concerned, CNNs have also been used in forest fire detections. The following table analyzes some studies on forest fire detection using CNN that have been used the last years.

Table 1: Relatively studies on CNN

Authors	Origin	Title	Method	Description
(Priya & Vani, 2019)	India	Deep Learning Based Forest Fire Classification and Detection in Satellite Images	CNN(Inception V3)	They used CNN based on Inception V3 approach to detect forest fires
(de Almeida Pereira et al., 2021)	Brazil	Active fire detection in landsat-8 imagery: a large-scale dataset and a deep-learning study	CNN(U-NET architecture)	They employ CNN based on U-Net architecture to detect active fires on a large-scale dataset
(Chen et al., 2019)	China	UAV Image-based Forest Fire Detection Approach Using Convolutional Neural Network	CNN, Local Binary Pattern (LBP), Support Vector Machine (SVM)	Chen et al. used an unmanned aerial vehicle (UAV) to detect forest fires. They first extract the feature using LBP and classify them using SVM to detect smoke. Finally, they apply CNN to detect forest fire.
(Hong et al., 2022)	China	Active Fire Detection Using a Novel Convolutional Neural Network Based on Himawari-8 Satellite Images	FireCNN	They used a new CNN method by using multi-scale convolution and residual acceptance design to detect fire spots
(Rostami et al., 2022)	Iran	Active Fire Detection from Landsat-8 Imagery Using Deep Multiple Kernel Learning	CNN(MultiScale-Net)	Rostami et al. applied a CNN MultiScale-Net to detect active fire on a pixel level.
(Georgiev et al., 2020)	Bulgaria	Forest Monitoring System for Early Fire Detection Based on Convolutional Neural Network and UAV imagery	CNN	They develop an object detection method based on CNN to detect forest fires using UAV

Finally, it is worth mentioning that the Inception V3 model which is implemented in this research is based on the structure of the CNNs. This research is using Inception V3 model because of the higher accuracy in small datasets as well as it needs less computing power to train the model and thus it has quicker results.

2.1 Cellular automata

Alan Turing's demonstration that computers could "regenerate" themselves marks the beginning of the history of cellular automata (CA), but the real development of this field rose in the middle of 1950s and 1960s (Crooks, 2017). The first person that employed CA was J. von Neumann in the theory of self-reproducing automata (Neumann, 1966) which he discussed the theoretical underpinnings of a machine that could replicate itself (Dewdney, 2018). However, the real creator was the mathematician Stanislaw Ulam (Bodrožić et al., 2006). One of the most famous models of CA is that of Conway's which is called "Game of life". The cells of this models possess only two states, dead or alive. Each cell evaluates the state of the nearby cells at the previous time step and the state of itself in the current time step, then it chooses one of three actions based on the results. For example, If the cell's present state is alive and the states of the other two cells in its immediate vicinity are also alive, the cell will continue alive, alternatively, it will die. Similarly, if the condition of three surrounding cells is alive, a cell that is now dead will turn to life. Finally, if the cell has three living neighbors or less than two alive neighbors, the cell will turn into dead (Crooks, 2017).

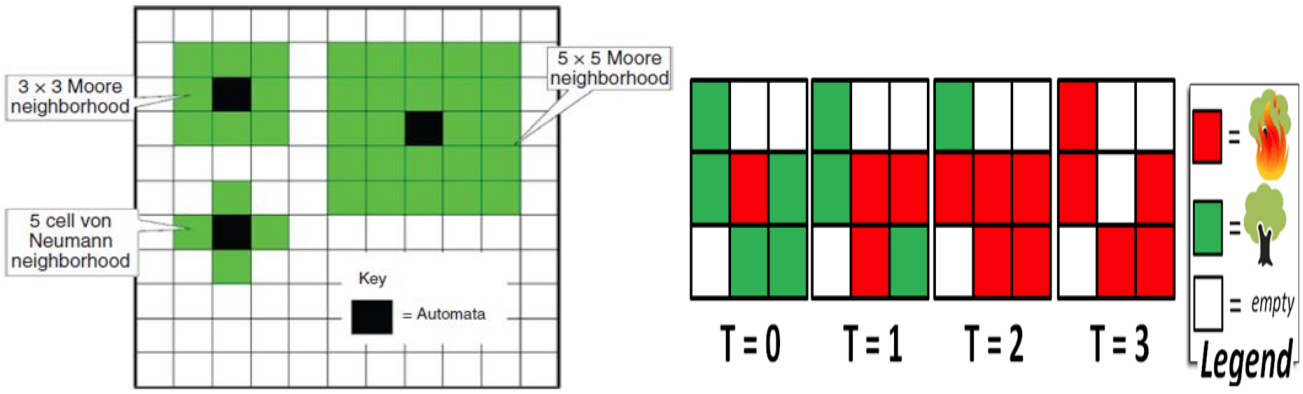
Many researchers have employed cellular automata (CA) models, due to its beneficial properties such as simplicity, adaptability, transparency, and intuitiveness (Zhu et al., 2021). According to Yang et al (2016) CA is a discrete-time grid dynamics model with local spatial interaction and temporal causation. Furthermore, this model could also simulate the spatio-temporal development of complex systems. Each cell's state varies over time and is characterized by local transition rules (as we saw in the Game of life), where the majority of the cells compose the evolution of dynamical systems by simple interaction (Yang et al., 2016). CA can be divided into four sections: the research area with a regular cellular entity, the cell state over a period of time, the interaction that a cell has with its neighbors, and given transition rules. In addition, the state of a cellular entity in the model at a given time is determined by the cell itself, its environment at an earlier time, and transition rules (X. Wu et al., 2021). Moreover, Dewdney (2018) gives the mathematical view of CA as representing in the equation below.

$$S_{x_{ij}}^{t+1} = f(S_{x_{ij}}^{t}, S_{\Omega_{ij}}^{t}) \qquad (1)$$

Where $S_{x_{ij}}^{t+1}$ and $S_{x_{ij}}^{t}$ describe the state of a cell, x_{ij} describes the location of the cell in the grid in a given time t and $t+1$ which they are a part of a limited set of cell states in the grid space. Ω_{ij} describes the state of the surrounding cells and $S_{\Omega_{ij}}^{t}$ refers to the state of the surrounding cells at a given time t and $t+1$. Finally, f is the function that describes a set of transition rules(Dewdney, 2018b).

As it mentioned earlier the state of a cell is directly connected to the state of neighboring cells. There are two common neighborhood configurations. The first one is called the 3x3 or 5x5 Moore neighborhood and the second one the von Neumann neighborhood (Bodrožić et al., 2006). The first takes the key cell in the middle and the 9 surrounding cells while the von Neumann neighborhood takes the key cell in the middle and the 5 surrounding cells. Another important aspect of CA is time. The figure below demonstrates in different time the state of cell. As can be seen the state of a cell varies over time based on different transition rules and the state of the surroundings cells.

Figure 4: On the left a 2-D cellular automata neighborhoods (Source: Crooks, 2017) and on the right the temporal aspect of Cellular automata (Source: Giabbanelli, 2019)

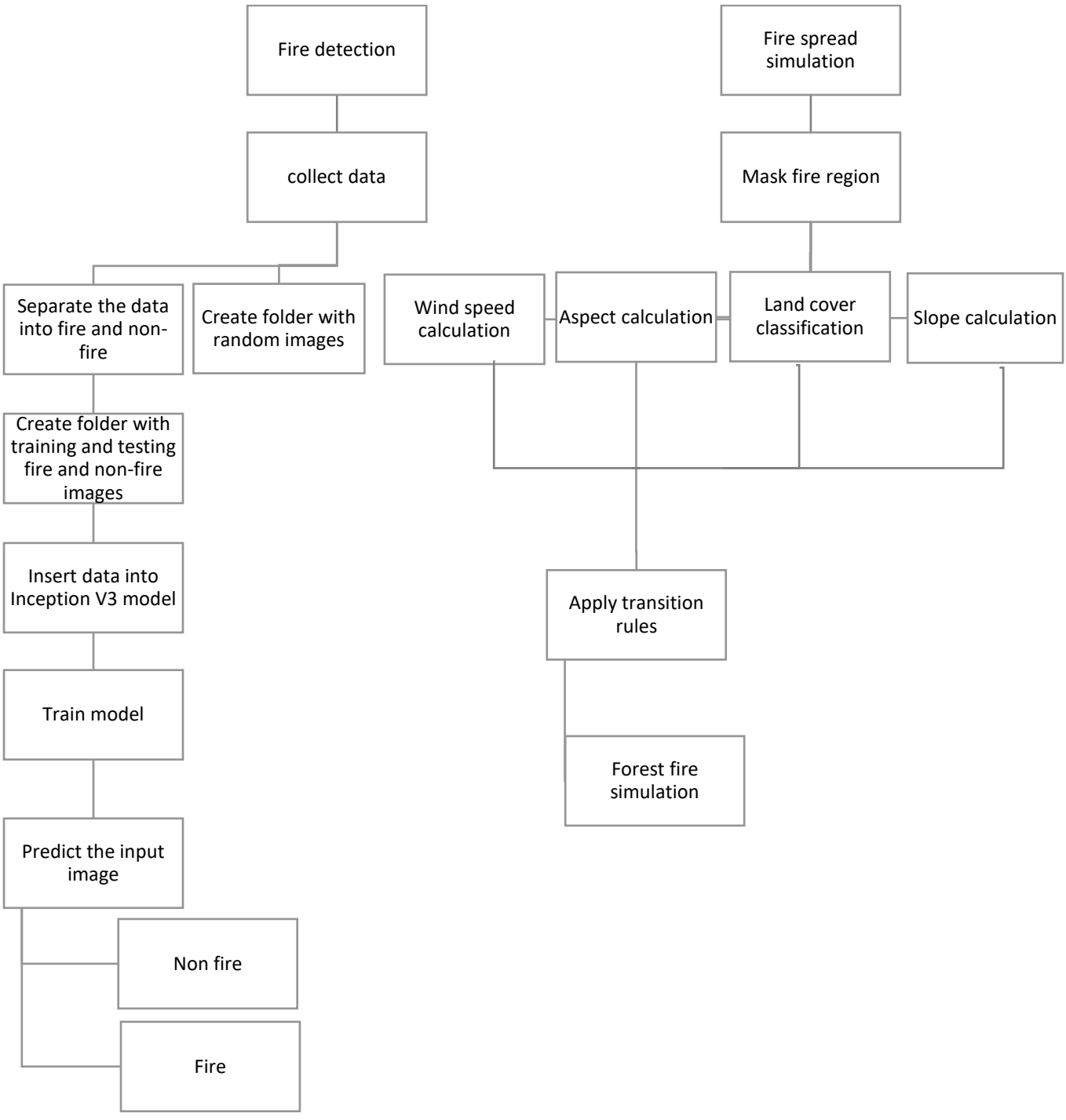


Over the last years the forest fires have significantly affect the forest ecosystems as well as the public safety, for that reason, research into simulating the spread of forest fires has significantly grown (Hernández Encinas et al., 2007). According to Alexandridis et al (2008) a model for predicting the spread of a wildfire would need to consider both individual topographical features and external environmental elements like weather conditions. (Quartieri et al (2010) mentioned that the amount of fuel and its type, the slope of the terrain, and the direction and speed of the wind are the most significant factors that affect how a fire spreads. Furthermore, Almeida and MacAu (2011) indicate that there are four distinct combustion phases that may be distinguished when the fire moves across the environment and consumes the vegetation. The first phase is the pre-heating, the second phase is the ignition, the third phase is the burning, and the last phase is the fire extinguishing.

3. Methodology

In this chapter the methodological framework is analyzed and explained. The figure below shows the steps of forest fire detection using the Convolution Neural Network as well as the steps for simulating the fire spread. As mentioned in the previous chapter, the result is intended to show whether an image has a fire or no fire in the area, and if there is a fire, it will show the area and the extent of the fire. After that, if the models shows that the image have fire, the cellular automata model is applied based on several transitions rules which are explained in this section.

Figure 5: Steps of fire detection and fire spread simulation.

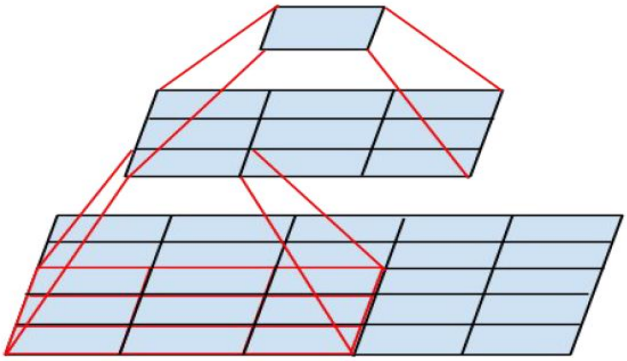


3.1 Inception-v3

As it mentioned in the previous chapter the scope of this research is to work on a CNN model to detect forest fire and simulate the course of a fire using CA. In this chapter, the Inception-V3 model is presented which is used as a CNN model to detect forest wildfires.

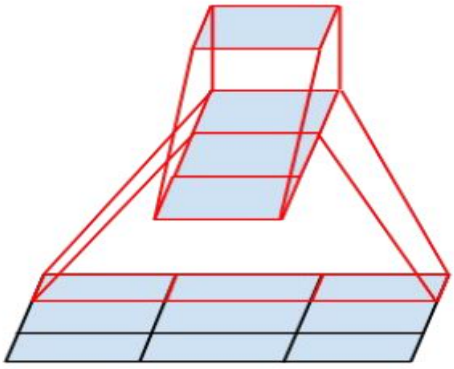
One of the main ideas of the Inception-V3 model which stands out from the rest is the Factorization into Smaller Convolutions. The main goal of this idea is to minimize the number of parameters without lowering network performance. Thus, the model can achieve faster training (Szegedy et al., 2016). In the figure below we can see an example of Factorization into Smaller Convolutions. The figure describes that two 3 x 3 convolutional layers can operate as one 5 x 5 convolution layer which has as a result to reduce the computing power and storage form 25 into 18.

Figure 6: A 3x3 convolution is replacing a 5x5 (Source: Szegedy et al., 2016)



In addition, as Szegedy et al (2015) mentioned convolutions with filters greater than 3 x 3 may not always be beneficial because they can always be converted to a series of 3 x 3 convolutional layers. However, 5x5 convolutional layer is not the only that can be divided into smaller layers. Szegedy et al. (2016) indicated that the layers could factorized into smaller by using asymmetric convolutions (figure 7). For example, a 3x3 layer could be divided into a 3x1 layer and then in a 1x3 layer. Moreover, they mentioned that in theoretical view this could expand even more as a n x n convolution may be replaced by a 1 x n convolution followed by a nx1 convolution, with a substantial rise in computational cost savings as n grows (Szegedy et al., 2016). However, this can only be implemented on a medium grid-sizes.

Figure 7: A 3x3 layer replaced by a 3x1 layer (Source: Szegedy et al., 2016)



The model that Szegedy et al. (2015) first introduced can be seen in the figure 8. This Inception model is the first model (Inception V1) of Szegedy et al. (2015) which represents four parallel layers. The main idea of this model was to capture various sizes of information in the image. However, this model is considered to be inappropriate due to the fact that

needs a lot of processing power which has a result to be time consuming. The next figure (figure 9) depicts the newer version of the inception model (Inception V3) which uses the factoring to a 3x3, a 3x1 and nx1 layer which results to less computing power and to a network with less dimensions and calculations that are completed more quickly.

Figure 8: Inception module without factorizing (Source: Szegedy et al., 2015)

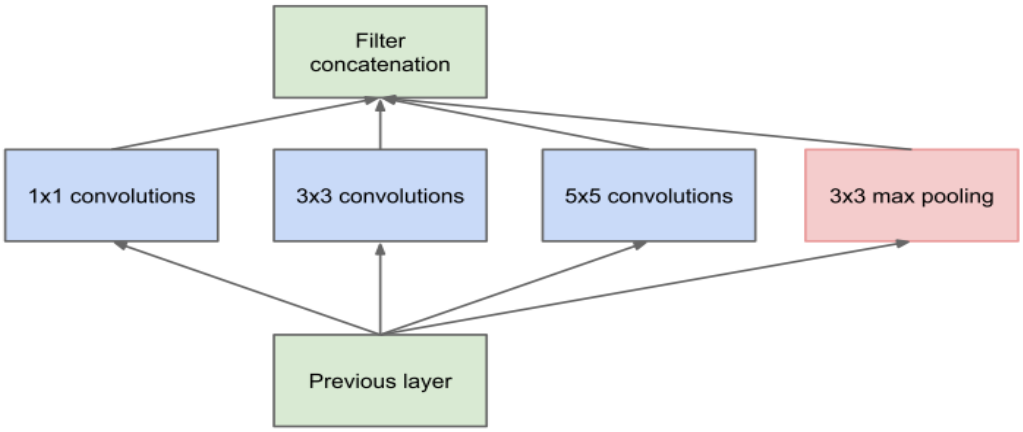
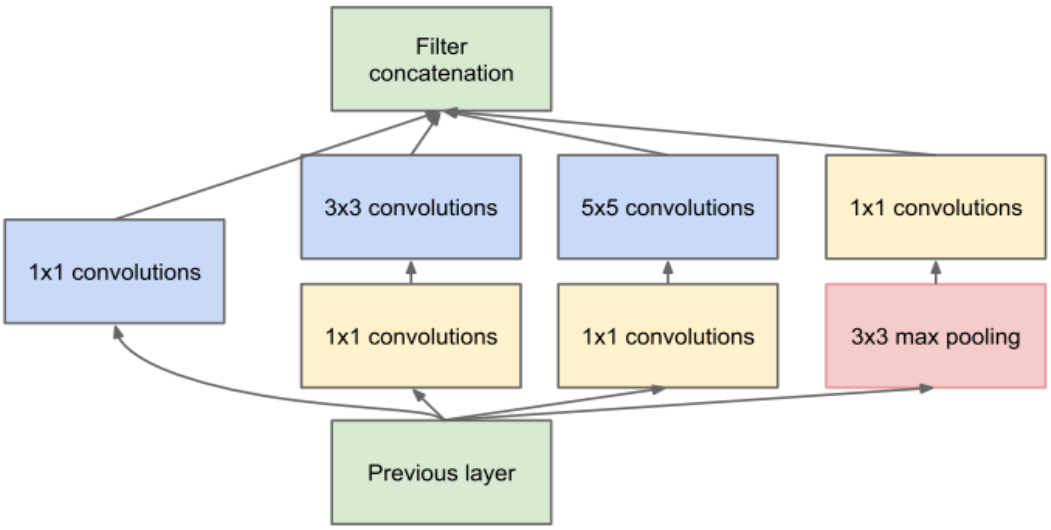


Figure 9: Inception V3 module (Source: Szegedy et al., 2015))

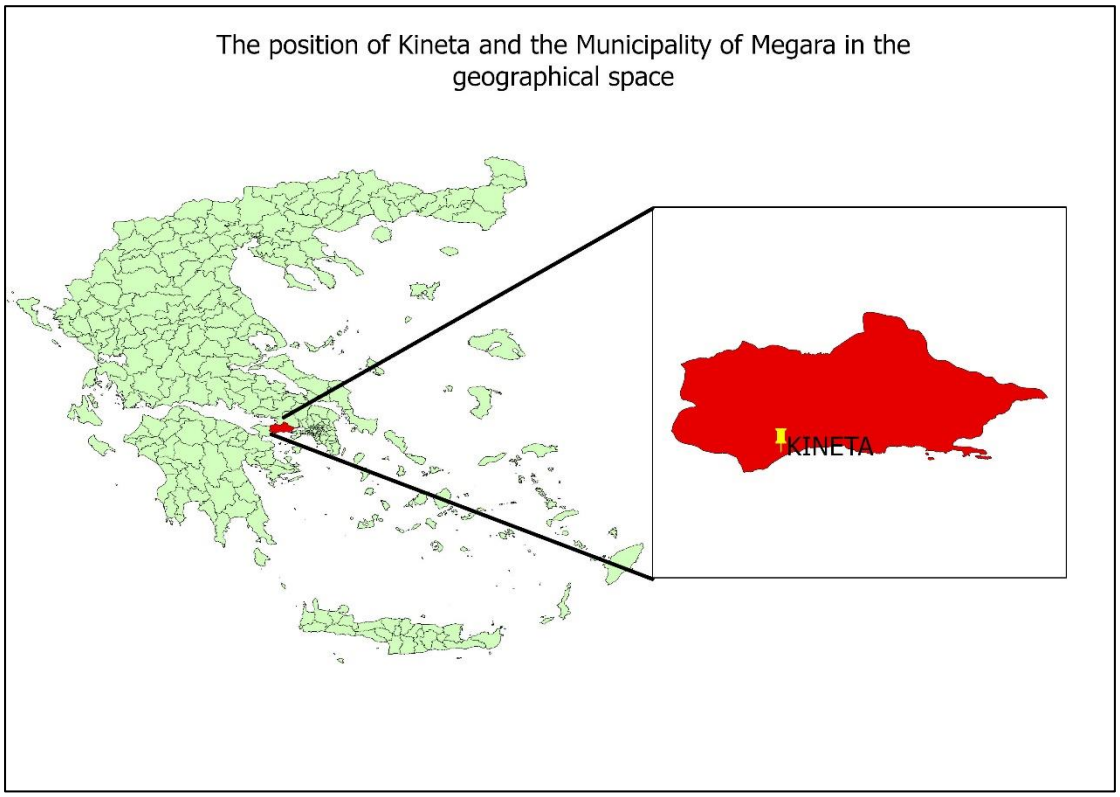


3.2 Study area and data collection

3.2.1 Study area

The study area of this research is located in the municipality of Megara in Eastern Attica in Greece near the city of Kineta. This area in 2018 experienced a great disaster from a forest fire by destroying large areas of forest and houses, as well as causing significant health problems to the people in the area. This forest fire and that in the municipality of Marathon in Western Attica in Greece near the city of Mati are the deadliest fire in the history of modern Greece. The study area will be used to evaluate the simulation model of forest fire spreading.

Figure 10: The location of Kineta in the Greek geographical space



Not far from the city of Kineta is the mountain of Geraneia with height over 1365m and it is considering the location where the fire started. Moreover, near the city of Kineta is the city of Agioi Thedoroi where this city and the city of kineta are considered important tourist areas. The figure below is a 3D model of our study area which depicts the area where the fire broke out as well as the two cities and the morphologies of the ground.

Figure 11: 3D model of Kineta area (Source: Google earth)

3.2.2 Data collection

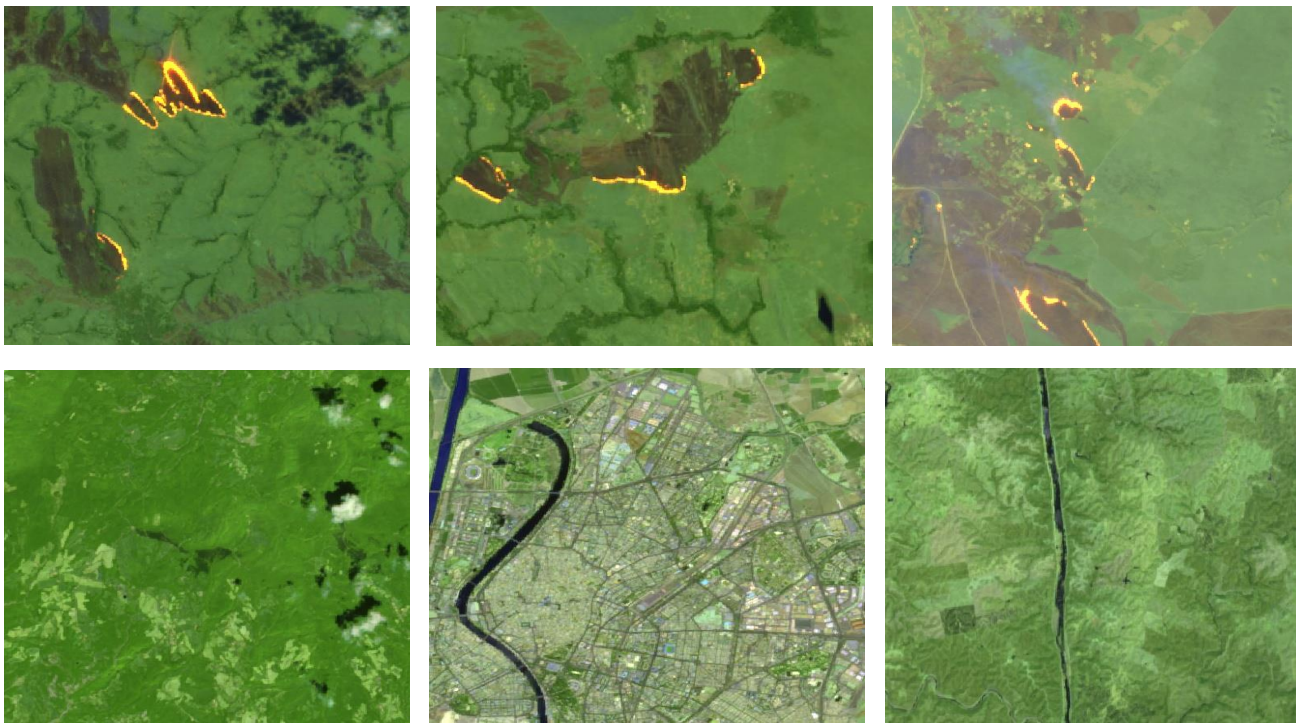
The data of this research was extracted from the Google Earth Engine using the Sentinel-2 MSI: Multispectral Instrument, Level-2A. First in order to identify the fire regions more easily the bands that was selected are B12(2100-2280 nm), B11(1565 – 1655nm) and B4(650-. 680 nm). The B12 band is in the shortwave infrared (SWIR2) as well as the B11 band in shortwave infrared (SWIR1) and the B4 is the red band. These bands were selected because B12 is substantially impacted by a burning fire and less vulnerable to ambient pollution than B4, which exhibits less susceptibility to fire (Hu et al., 2021b). Furthermore, B12 and B11 shows high surface reflectance in the fire region making it easier to detect by naked eye (Hu et al., 2021b). The figure below shows the difference between a true color sentinel-2 image and a B12, B11, B4 combination of a sentinel-2 image in a fire region. These two images are from the same sensor, and they capture the same time.

Figure 12: On the left is a B12, B11, B4 combined image and on the right a true color image.

According to the figure above it can be seen that the combination of these bands captures the fire area very accurate. In contrast, the true color image doesn't give the exact location of the fire region. Moreover, the data were selected from different areas around the world were sentinel-2 satellite was available. This was done to train the model with fire and non-fire images, since in our study area there is not much data of fire images. In addition, in order to train the model into different environments and climates data from around the world must be added.

The data was downloaded with the three bands that are mentioned earlier and they were separated into two categories 'fire' and 'Non fire'. After that they separated into train and testing. The images that contain fire are 146 and the images that are not fire are 118. The images that were used for train are 198 and 75 for testing. Furthermore, 27 random images were also selected that are not in the model in order to check the accuracy of the model on random images.

Figure 13: Samples of the sentinel-2 images. On top are the fire images and, in the bottom, the non-fire images.



Finally, the data that was used to simulate the fire spread are mentioned in the table below. The data are focused in the area of Kineta and they was used to formulate the transition rules.

Table 2: Sources of data

Data	Type (spatial resolution in meters)	Source	Date
Sentinel-2 MSI LEVEL 2A	Raster grid (20x20 m)	ESA Copernicus https://scihub.copernicus.eu	23-07-2018
Sentinel-2 MSI LEVEL 2A	Raster grid (20x20 m)	ESA Copernicus https://scihub.copernicus.eu	14-08-2018
Corine Land Cover	Vector	ESA Copernicus	2018

(CLC)		https://scihub.copernicus.eu	
Average wind speed(km/h)	Data table	Meteo http://stratus.meteo.noa.gr	23-07-2018
DEM	Raster grid (25x25 m)	ESA Copernicus https://www.copernicus.eu/en	01-11-2022

The first sentinel-2 image was captured the day that the fire was started, and the second sentinel-2 image was captured after three weeks because the cloud percentage was very low. Moreover, this image was also used to identify the burned area and see how well the model simulated the fire spread. Finally, the Corine land cover data was used to identify the land cover in the area of Kineta.

3.3 Land cover classification

In order to simulate the spread of the fire it is important first to extract the fire region the time the fire started. For that reason, the Support Vector Machine (SVM) was used to mask the fire region. The SVM is a supervised algorithm which uses training samples to assign labels to objects(Noble, 2006). After that the fire region was implemented into the land cover data.

In order to simplify and fit the model's purposes, similar land cover categories were grouped into the same class with the final result be with 6 classes, 1) Fire, 2) High density forest, 3) Low density forest, 4) Barren soil, 5) Human structures, 6) Water. The table in the appendix I describes the reclassification of the data in more details. The total amount of cell of each class is depicted in the table below. For this study the selected cell size is 25 x 25. Finally, the result of the reclassification can be seen in the figure 14.

Figure 14: Number of cells for each category

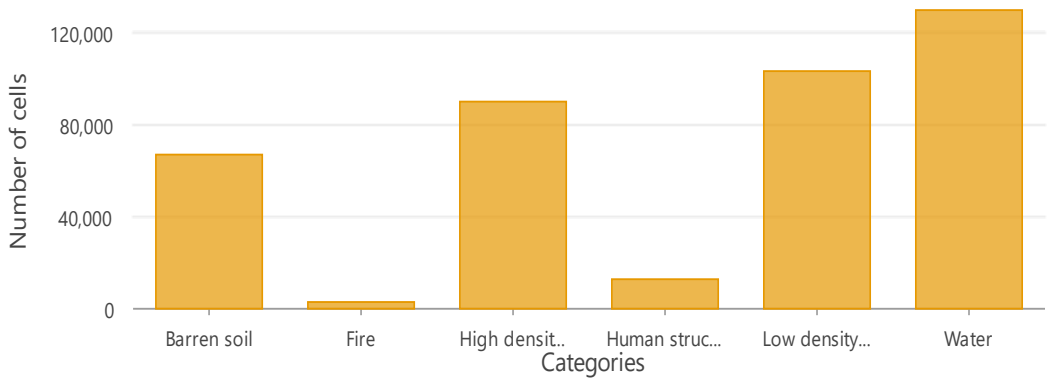
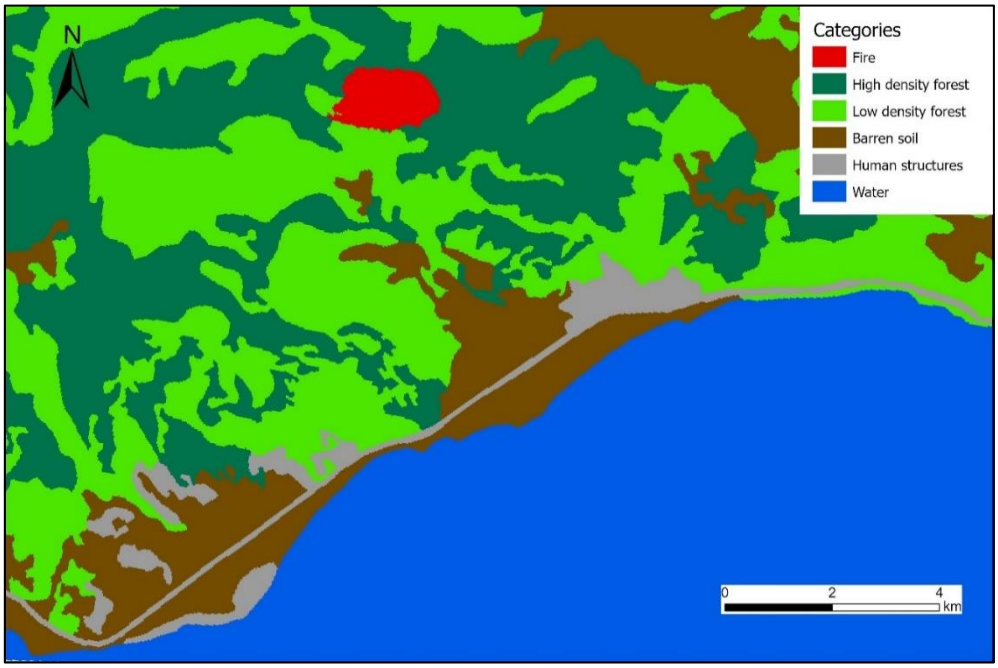


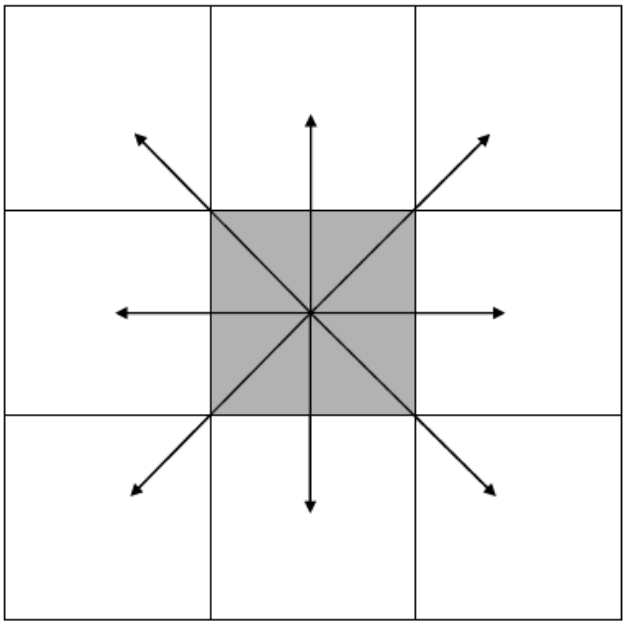
Figure 15: Land cover classification in Kineta



3.4 Fire spread model

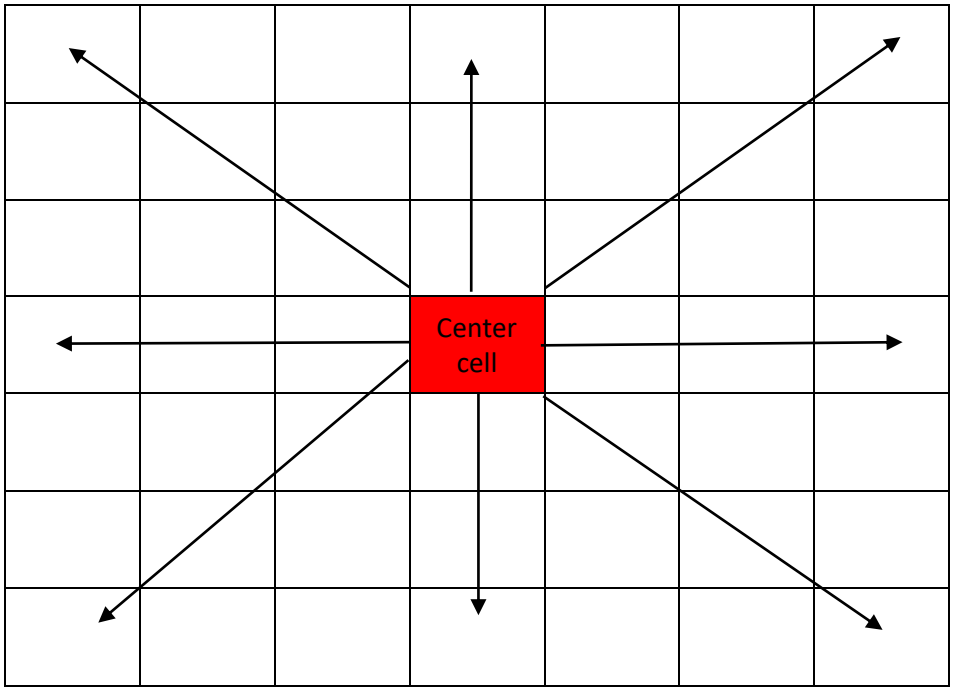
The method divides the study area into a number of cells using a two-dimensional grid. Each square cell represents different land cover type which provides eight potential paths for the development of fire, as can be seen in the figure below.

Figure 16: Possible direction of fire expansion (Alexandridis et al., 2008).



According to Alexandridis et al. (2008), many researches applied hexagonal cells and even though they can capture the spatial behavior of the fire more accurately, the computing complexity of the resulting model is greatly increased. In this research in order to simplify the model the square grid behavior is utilized which the neighborhood of the center cell is 75x75 meters on each direction, i.e., 3 cells in each direction. This can be better depicted in the figure below.

Figure 17: Cell neighbourhood



Each time step the evolution cells state is depicted by a set of rules which can imitate the behaviour of the fire propagation. Generally, a cell will be become burned the next time step if it meets all the transition rules. Furthermore, this research uses a propagation model which was already used by Alexandridis et al. (2008) and it can be described in the equation below.

$$P_{burn} = P_h(1 + P_{den})(1 + P_{veg})P_w P_s \qquad (1)$$

Where P_h represents the (constant) likelihood that, in the absence of wind and on a flat surface, a cell holding a specific type of vegetation and a given density of vegetation will catch fire at the following time step(Alexandridis et al., 2008), and P_w, P_s , P_h, P_{den} and P_{veg} are the probabilities of cell catching on fire which depends on the density of the vegetation, the type of the vegetation, the wind effect and the effect of the terrain. Finally, in order to produce the corrected probabilities that accounts all the aforementioned elements the P_h constant has been added which multiplies all the variables(Alexandridis et al., 2008).

3.4.1 The effect of wind

Wind has an important role on forest fire spread and it highly influence the route of the forest fire as well as the fire propagation. In addition, Wind availability and wind speed are mostly influenced by local terrain and weather patterns (Angeles-camacho, 2021). Unfortunately, in our study area only one meteorological station is located close to the spread of the fire and the data were based on these measurements. According to Pintor et al. (2022) there are two methods that have been widely used to extrapolate wind speed, the first method employs a distance calculation that takes into account all measurement heights, and the second method is by using wind shear coefficient. The fist method is the logarithmic profile, and the second method is the power law. Furthermore, their study showed that the second method has better results than the first method and is therefore more reliable. In this research the second method was selected.

In order to calculate the wind speed for each height, the wind shear coefficient must be determined. According to Erdik and Law, (2012) the wind shear coefficient ranges between 0.14 and 0.2. However, the coefficient is not constant as it is affected by a variety of variables such as atmospheric conditions, temperature, pressure, humidity, time of day, season, average wind speed, direction and soil type (Erdik & Law, 2012). This research uses values of the wind shear exponent for different types of terrain based on the Engineeringtoolbox, (2020) which can be seen in the table below.

Table 3:Wind shear for various terrain types (Engineeringtoolbox.com, 2020)

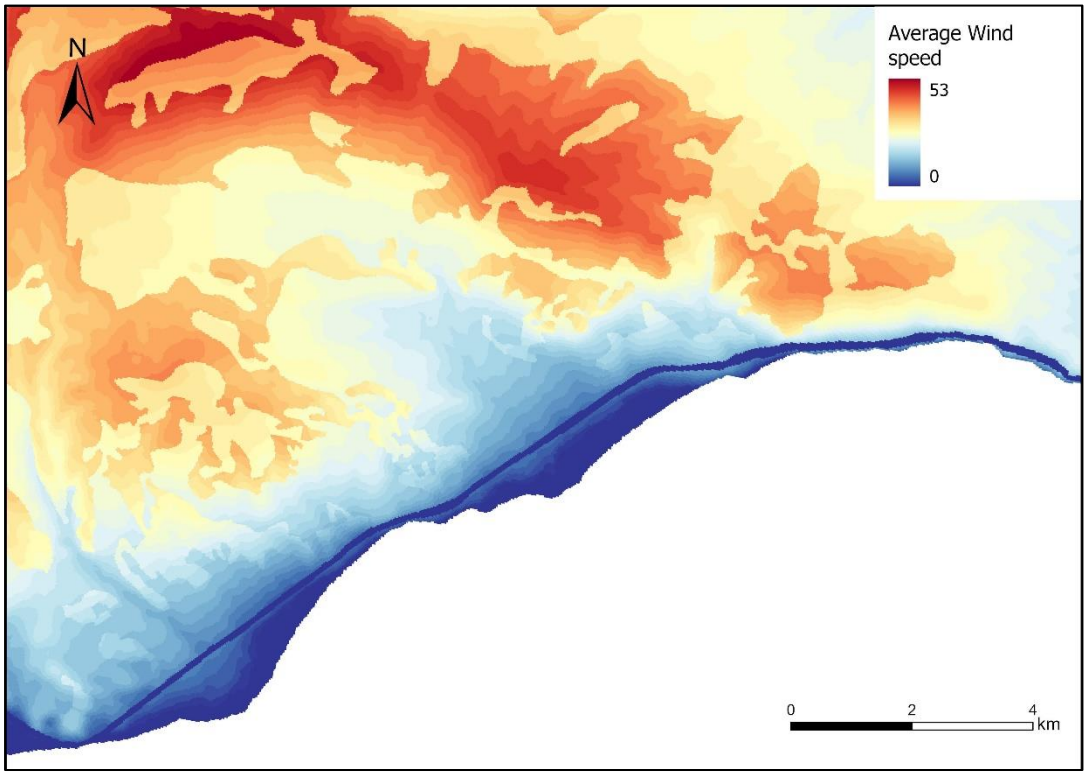
Terrain	**Wind shear exponent (a)**
Open water	0.10
Smooth, level, grass-covered	0.15
Row crops	0.20
Low bushes with a few trees	0.20
Heavy trees	0.25
Several buildings	0.25
Hilly mountainous terrain	0.25

The meteorological station that was near the fire spread has data about the wind speed and direction every 15 minutes. For that reason, the average wind speed was calculated from 09:00 a.m. to 00:00 p.m. to depict more accurately the changes in the wind speed. Furthermore, 09:00 a.m. is considered the time when the fire broke out and it lasted less than 20 hours. For the sake of this research, we consider that the fire lasted from 09:00 a.m. till 03:00 a.m., that is 18 hours. The station is situated 7km from the point where the fire broke out and the average speed was 20 km/h and the elevation is 37m. The power law equation depicted in the equation below was used to determine the average wind speed for each height in the Kineta area.

$$U_{average} = U_{ref} x \left(\frac{h}{h_{ref}}\right)^{a} \quad (2)$$

Where $U_{average}$ is the average wind speed for every height, U_{ref} is the referenced average wind speed, h is the new height, h_{ref} is the referenced height and a is the wind shear coefficient which is depicted in the table above. This research matched the land cover type with the different type of terrain. The fire the human structures and the high-density forest is equal to 0.25. The low-density forest and the barren soil are equal to 0.20 and the water is equal to 0.1. The results of the equation can be seen in the figure below.

Figure 18:Average wind speed in Kineta.



As can be seen in the figure above high values of wind speed are concentrated at high altitudes where high-density and low-density forest are in the area. In contrast, low values of wind speed are near cities and in barren soil. After that, the effect of the wind for each cell was calculated based on the equation below(Mutthulakshmi et al., 2020).

$$f_t = e^{VC_2(\cos\Theta - 1)} \quad (3)$$

$$P_w = e^{C_1 V} f_t \quad (4)$$

Where V is the average wind speed for each cell, θ is the angle between the direction of the wind and the direction of fire propagation. The wind direction at the beginning of the outbreak of the fire was south west and the direction of fire propagation was south east, hence the angle direction is 90°. In addition, the c_1, c_2 are constant parameters which were adopted by Alexandridis et al. (2008) and can be seen in the table 5. It is worth mentioning that the wind direction in our study area was Southwest when the fire broke out, however during the fire spread the wind direction changed several times, this will be depicted in the aspect parameters later on. Furthermore, when the wind direction and the fire direction are aligned, then the rate of fire propagation rises(Mutthulakshmi et al., 2020).

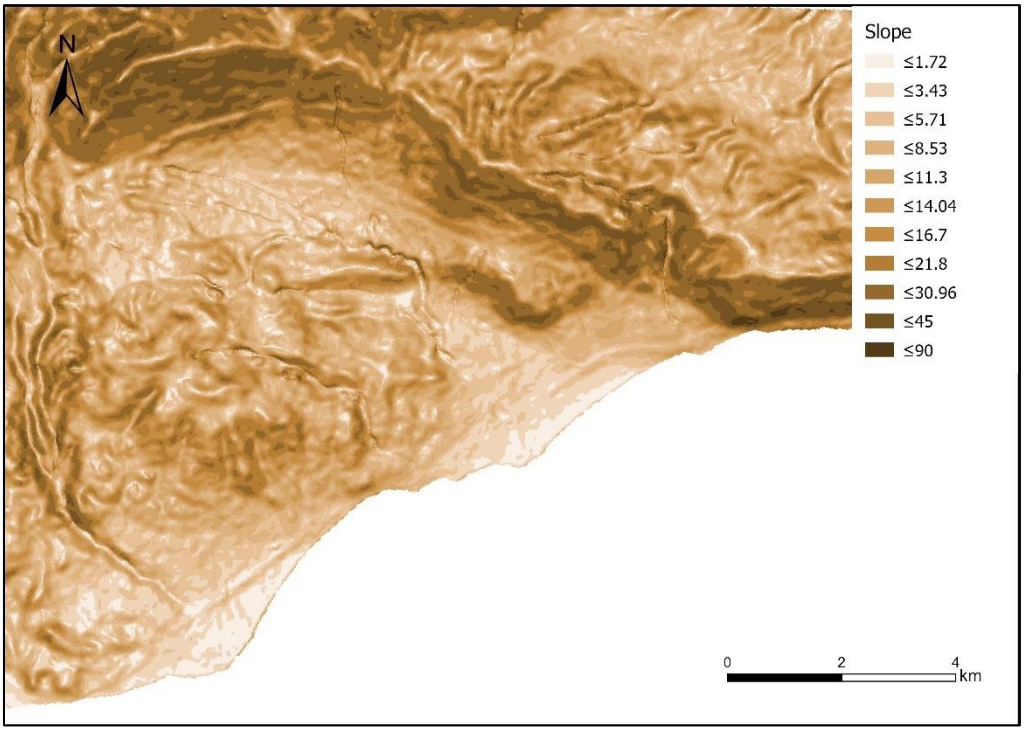
3.4.2. The effect of terrain

Terrain also has an important role in the fire propagation. For that reason, this research takes it into account. The equation for the effect of the terrain can be seen in the equation below(Freire & Dacamara, 2018).

$$P_s = e^{a_s \theta_s} \quad (5)$$

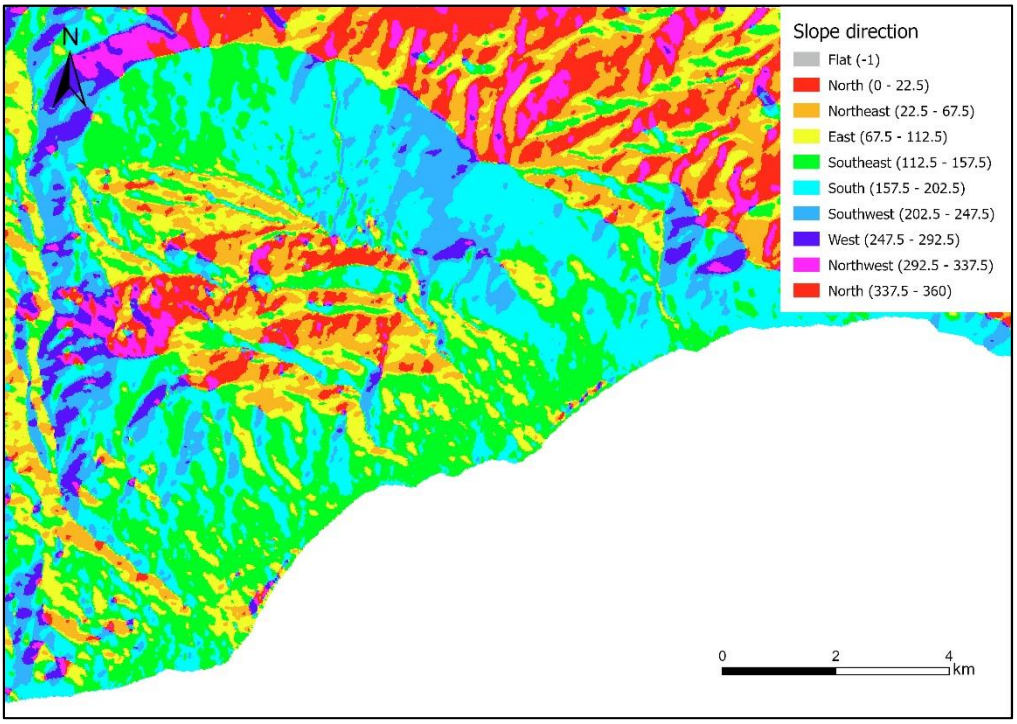
Where θ_s is the terrain's slope angle and a_s is a parameter that may be changed based on experimental results. The slope values of our study area are depicted in the figure below.

Figure 19: Slope values in Kineta



The figure shows high values of slope in the mountains of Geraneia and as it goes in lower altitudes the slope decreases and stabilizes. In addition, this research also uses aspect as a parameter that affects fire spread which as can be seen later it has been used in the transition rules. Generally, when the wind direction and the aspect are aligned then the fire propagation increases. In our study area, the wind direction between the outbreak of the fire and until 12 in the evening, the wind direction was Southwest, West, South, Southeast, East. For that reason, the fire spread is increased in these directions of the slope, as can be seen in the figure below.

Figure 20: Aspect values in Kineta



3.5. Transition rules

As mentioned in the previous section this research takes into account various parameters in the fire spread model. However, there are some parameters that do not directly affect or have minimal effect on the fire propagation model. For this reason, some assumptions are made at the beginning of the model. These assumptions are related with the land cover type and are analyzed in the table below.

Table 4: Land cover assumptions

Land cover type	Assumptions
High density forest	Fire can move through the high-density forest more easily because the fuel is higher.
Low density forest	Fire can move through low-density forest at moderate speed because the fuel is thinner
Barren soil	Fire cannot move through the barren soil type because there is no fuel to burn.
Human structures	Fire cannot move through the Human structures type because there is no fuel to burn.
Water	Fire cannot move through the water because there is no fuel to burn.

After these assumptions, the equation (1) was c Values of $P_{vegetation}$, $P_{density}$ probabilities and constant parameters calculated in the land cover types that can change. The constant parameters that were used are depicted in the table below.

Table 5:Values of $P_{vegetation}$, $P_{density}$ probabilities and constant parameters

Land cover type	$P_{vegetation}$
Mixed-coniferous Forest	0.4
Natural grasslands-Sclerophyllous Vegetation-Transitional woodland-scrub	0.2
Land cover density	$P_{density}$
High density forest	0.3
Low density forest	0.2
Constant parameters	values
P_h	0.58
c_1	0.045
c_2	0.131
a_s	0.078

The majority of these values were from Alexandridis et al. (2008) while some values of the low density forest changed to depict more accurately our study area. After defining every variable, the final transition rules that were applied in the fire spread model can be seen in the table below. It is worth noting that the transition rules are applied to each cell before beginning of the period. Furthermore, in our simulation scenario every *'ticks'* symbolize 15 minutes period.

Table 6: Fire spread conditions.

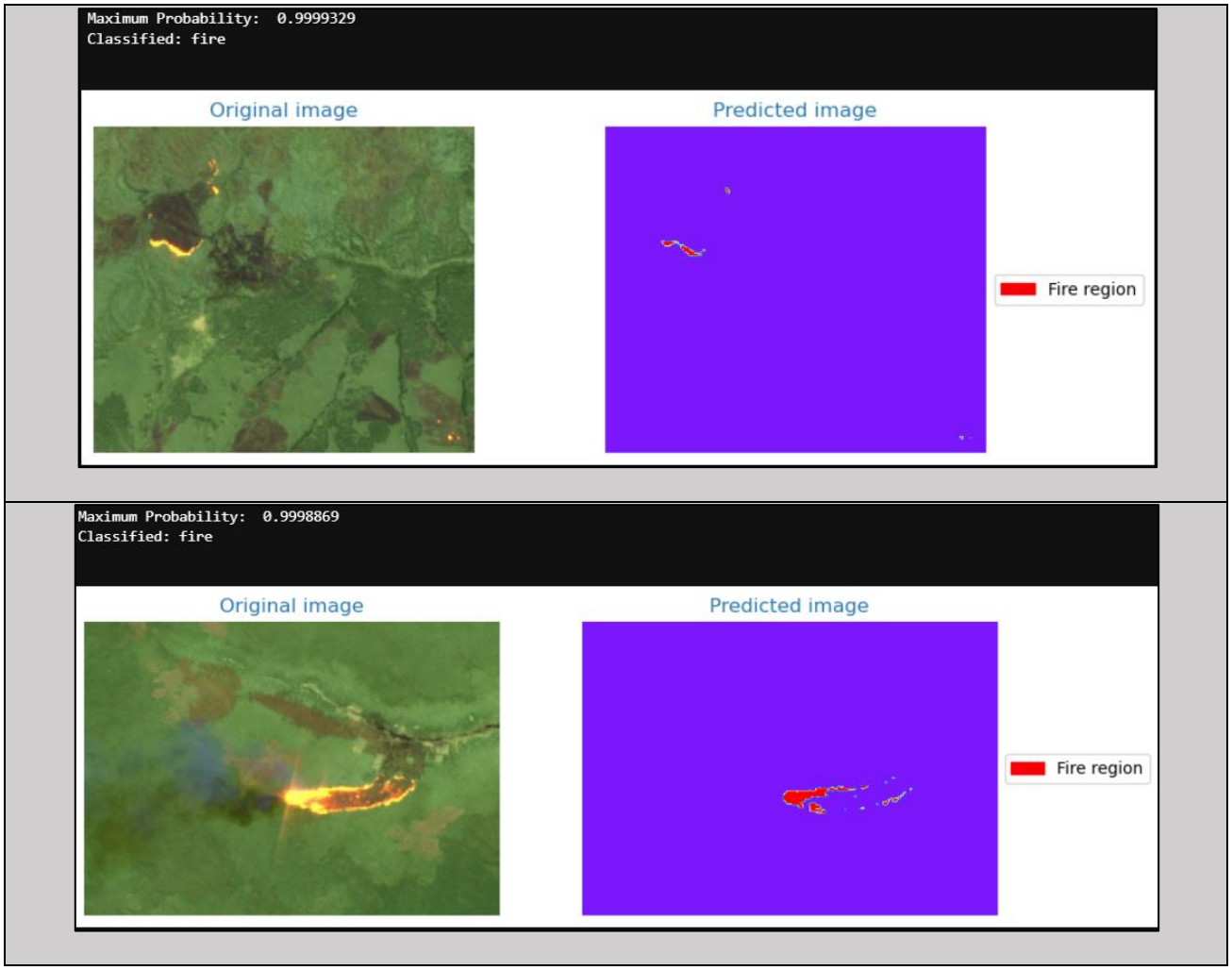
If the probability of the high-density forest is greater than 0,018 **AND** the total amount of high-density forest neighbors is greater than the low-density forest neighbors **AND** the aspect is East, Southeast, South, Southwest, West.	Fire birth
If the probability of the low-density forest is greater than 0,013 **AND** the total amount of low-density forest neighbors is greater than the high-density forest neighbors **AND** the aspect is East, Southeast, South, Southwest, West.	Fire birth
If the above conditions are not met, no cell will be changed to fire.	Land cover type

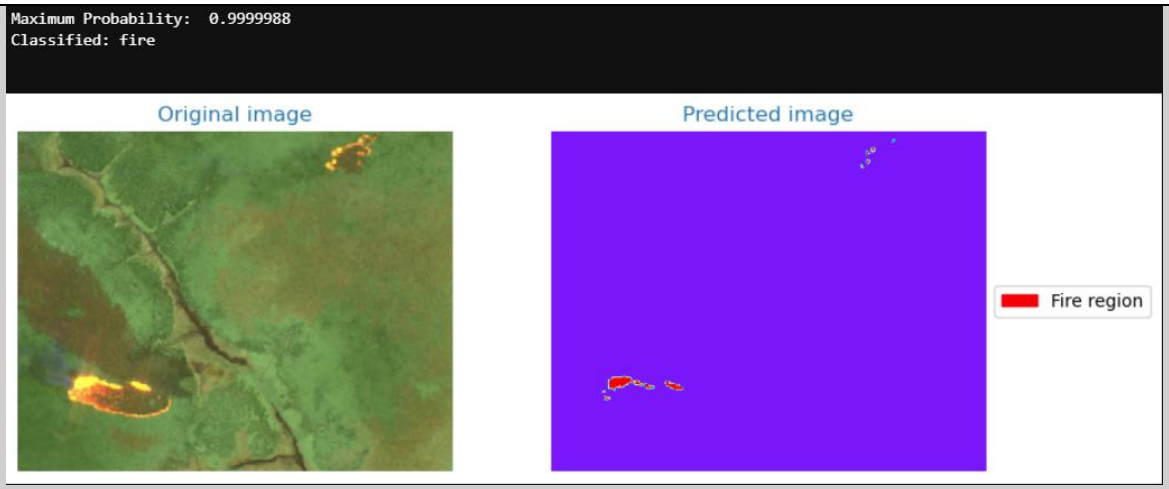
4. Results

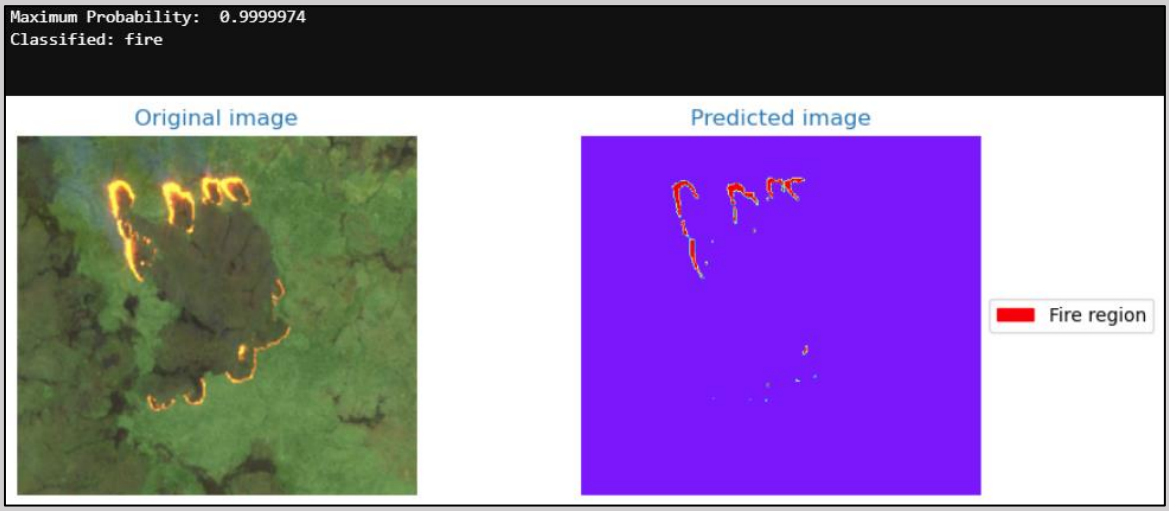
4.1 Forest fire prediction

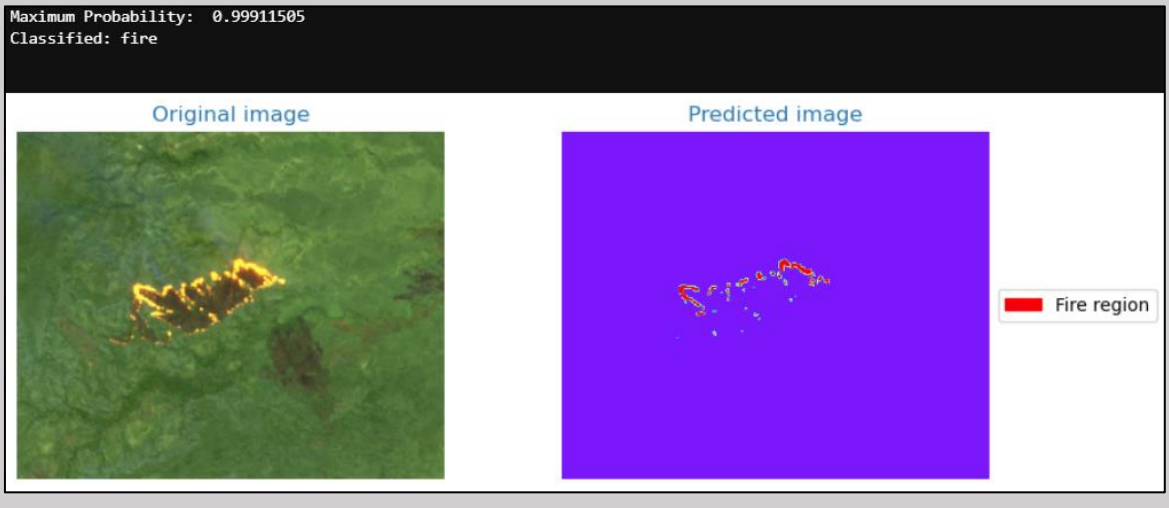
As mentioned before the first step of this thesis is to detect forest fire spread by using machine learning. After running the inception V3, which trained the model, random images were selected to identify if the model can recognize fire and non-fire images. Even though, the dataset is not very large the model is well detecting the fire and non-fire image as can be seen in the figures below. Moreover, the procedure for fire detection is described in the appendix iii as well as the graphs of validation and training accuracy.

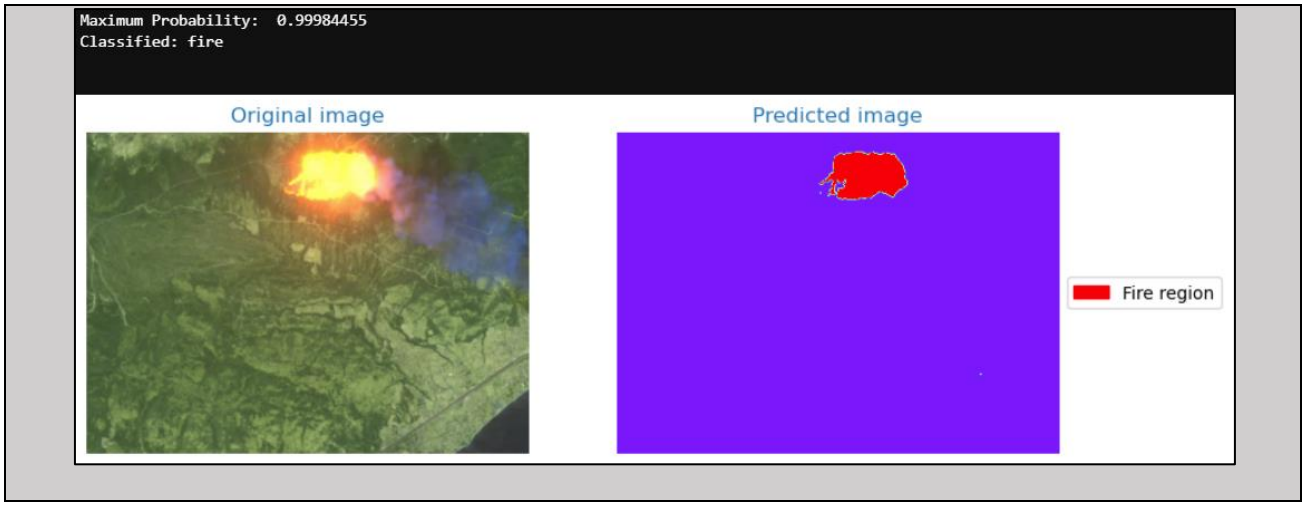
Figure 21: Fire detection




Maximum Probability: 0.9999988
Classified: fire
Original image
Predicted image
Fire region


Maximum Probability: 0.9999974
Classified: fire
Original image
Predicted image
Fire region

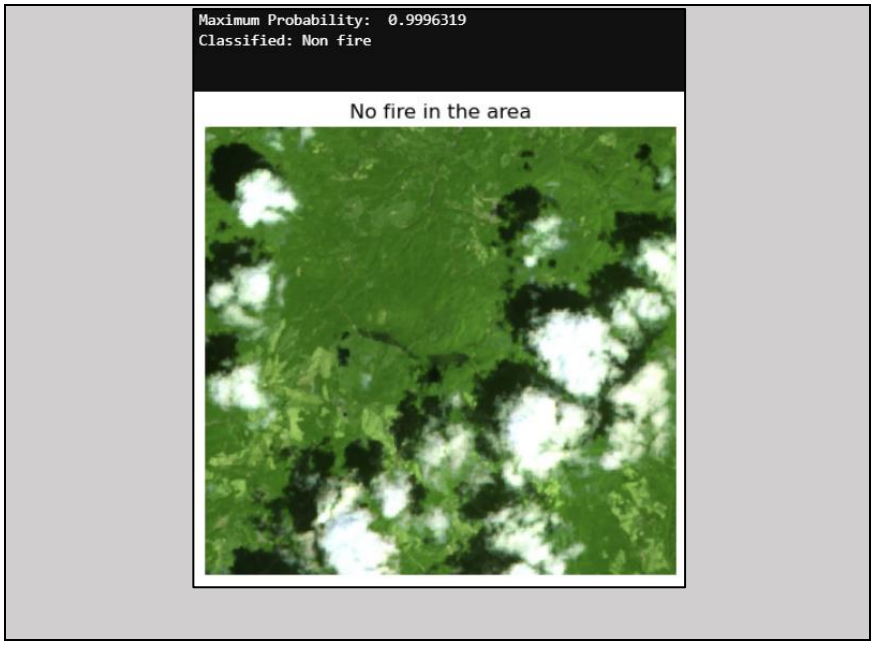

Maximum Probability: 0.99911505
Classified: fire
Original image
Predicted image
Fire region



The figure above shows the results of fire detection on six different images. The images are separated into the original image with the B12, B11, and B4 band combinations and the image with the fire region mask. The results show that the inception V3 model is capable of detecting fire images and masking the fire region. At this point, it is worth mentioning that the mask of the fire region was performed using the cell values; for that reason, the accuracy is not considered optimal, but it gives a very good representation of the area of fire. Furthermore, the maximum probability describes the percentage of confidence that the image has fire. The results indicated that the confidence level of fire existence was over 99%, which signifies that the images have fire.

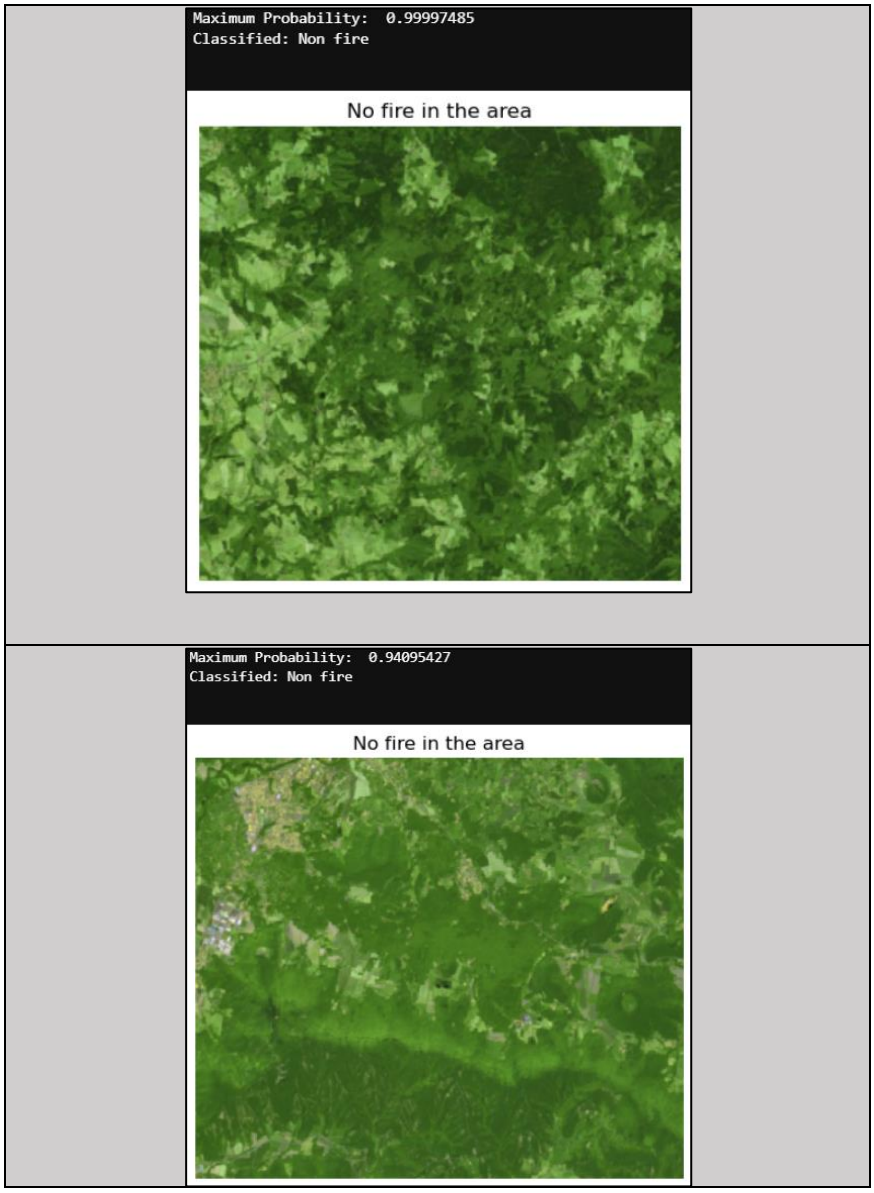
When it comes to non-fire images, the results only show the original images because they do not contain any fire. The results of the non-fire images can be seen in the figure below.

Figure 22: Detection of Non fire images



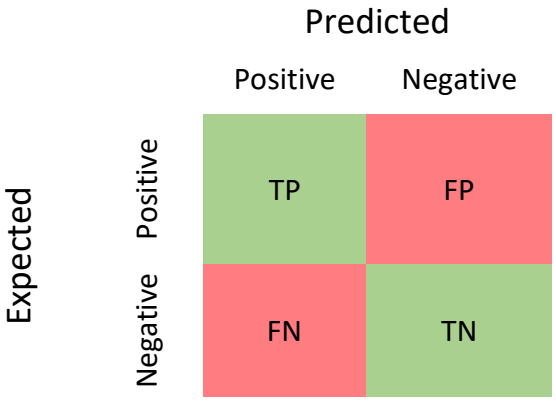

Maximum Probability: 0.9732152
Classified: Non fire
No fire in the area


Maximum Probability: 0.99711215
Classified: Non fire
No fire in the area



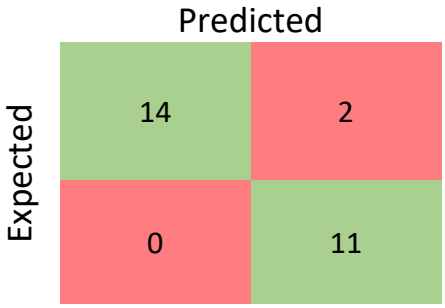
According to the results above, the inception V3 model detects the images that do not contain fire quite well. The maximum probability is also over 99% except for the images that have urban fabric. In order to validate the model, 27 random images were extracted that were not included in the training or testing. The evaluation of model was conducted using the Confusion Matrix. This tool evaluates the performance of the model. A typical confusion matrix has True Positive values (TP), True Negative values (TN), False positive values (FP), and False Negative values(FN)(Ting, 2017). In our example the TP values represent the fire images and the images which the model predicted correct. In the TN values show the non-fire images and images which the model predicted correct. The FP values indicate the fire images which the model predicted incorrectly. Finally, the FN values represent the non-fire images that the model predicted as fire. The figure below illustrates the confusion matrix.

Figure 23: Confusion Matrix



The results of the model are depicted in the figure below. The figure shows that all the fire images were well predicted from the model whereas from the non-fire images two were predicted as fire.

Figure 24: Confusion Matrix of the model



From the 27 random images the 14 had fire and the other 13 they didn't. The total accuracy of the model reaches 92.5%, which is very good given the fact that the data was not very large. In addition, the F1-score was calculated in order to evaluate further the accuracy of the model. This model incorporates the results of the recall and precision. Generally, values closer to 1 indicate that the model have high precision and recall and closer to 0 that the model has low precision and recall. The recall and the precision can be calculated with the following equations.

$$Prec = \frac{TP}{TP+TN} \quad (6)$$

$$Recall = \frac{TP}{TP+FN} \quad (7)$$

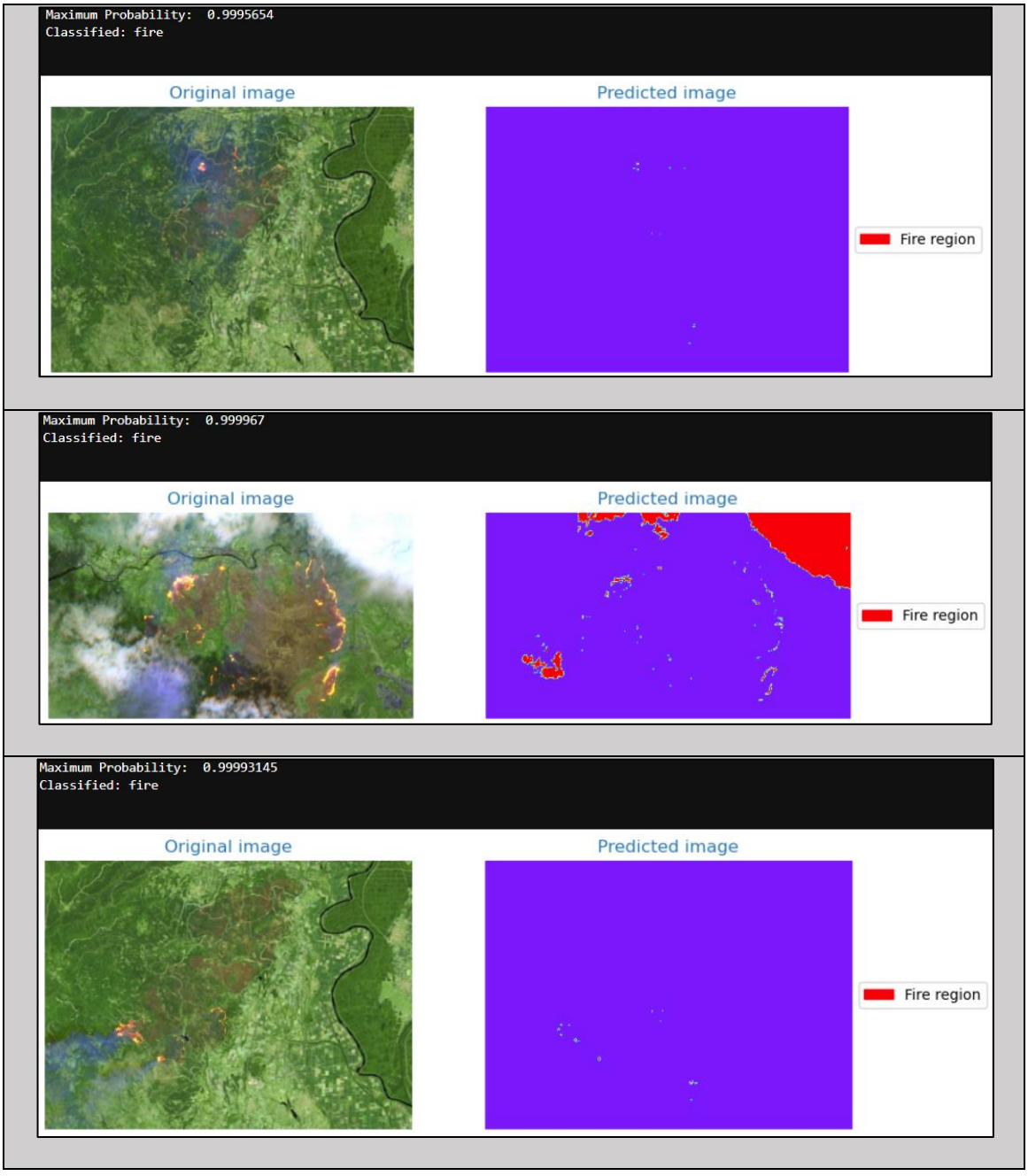
The recall calculates the fire images that the model was able to predicted correctly while the precision shows how many of the total correctly predicted images proved to be fire images(Ting, 2017). After the calculation of precision and recall the f1-score was calculated based on the equation below(Goutte & Gaussier, 2005).

$$F1 - Score = 2 * \frac{Prec * Recall}{Prec+Recall} \quad (8)$$

The F1-score is approximately 0.72 which shows that the model accuracy based on the precision and recall is very well.

Finally, in order to answer the research questions that were posed at the beginning, the model was also tested on larger images. The aim of this was to identify if the image size affected the performance of the model. For that reason, three images that contain fire were selected as well as two images that do not have fire. The results of the prediction are depicted in the figure below.

Figure 25: Testing on larger images

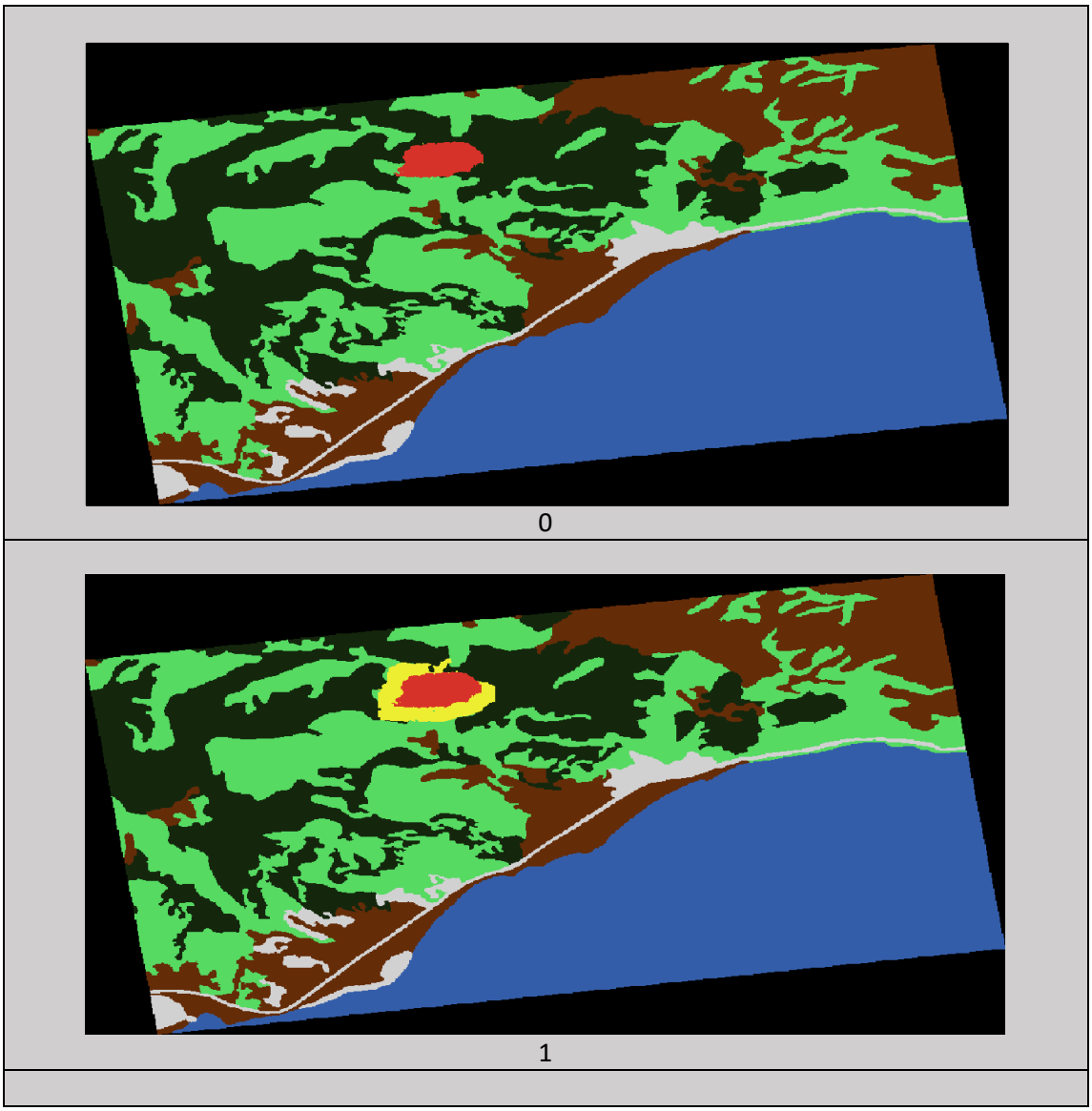


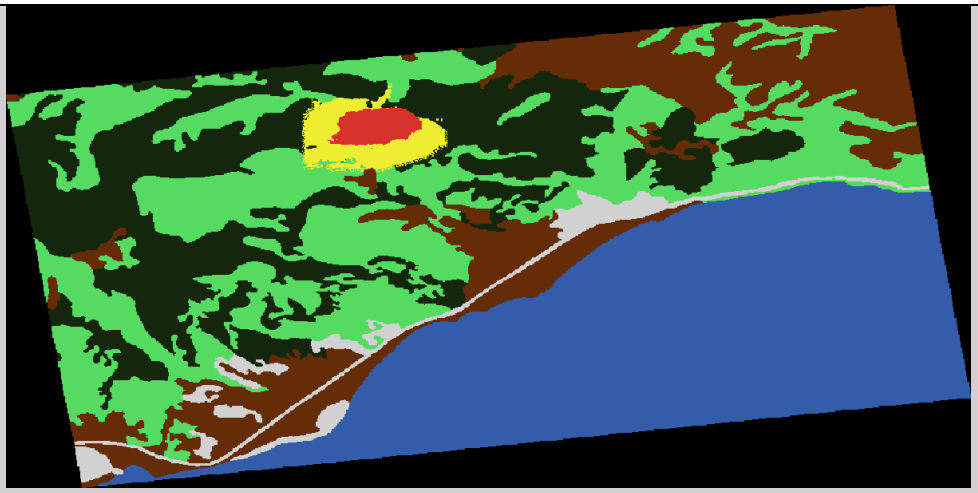
The model's output demonstrated that it can distinguish between pictures of fire and non-fire on a broader scale. However, the mask of the fire region doesn't depict very well. Moreover, the mask of the fire region capture clouds as fire this is because the mask is based on the pixel value and not on image classification. Taking everything into consideration, the model performs very well on detecting the possibility of fire on small and larger images. However, when it comes to masking the fire region, it needs further investigation because the fire region based on the cell value doesn't give very accurate results.

4.2 Forest fire spread simulation

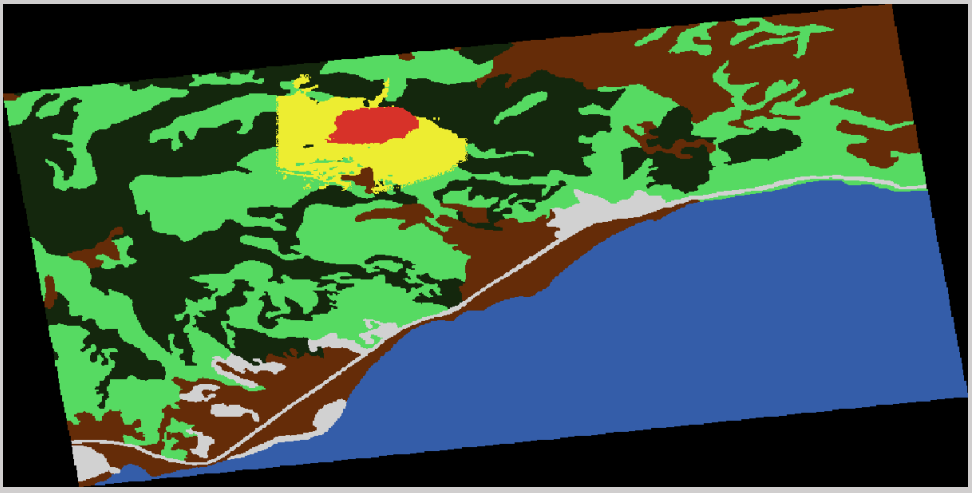
The second stage of the thesis is to simulate the spread of a forest fire. For that reason, the Kineta area, as described in the previous chapter, was chosen, which suffered significant damage from the wildfire that erupted in 2018. The Appendix iv describes the process that this thesis followed in order to produce the model. The result of the simulation starts from 0 hours which is consider when the fire broke out and continues for 18 hours. The figure below shows the outcome of this simulation.

Figure 26: Fire spread simulation each time step.

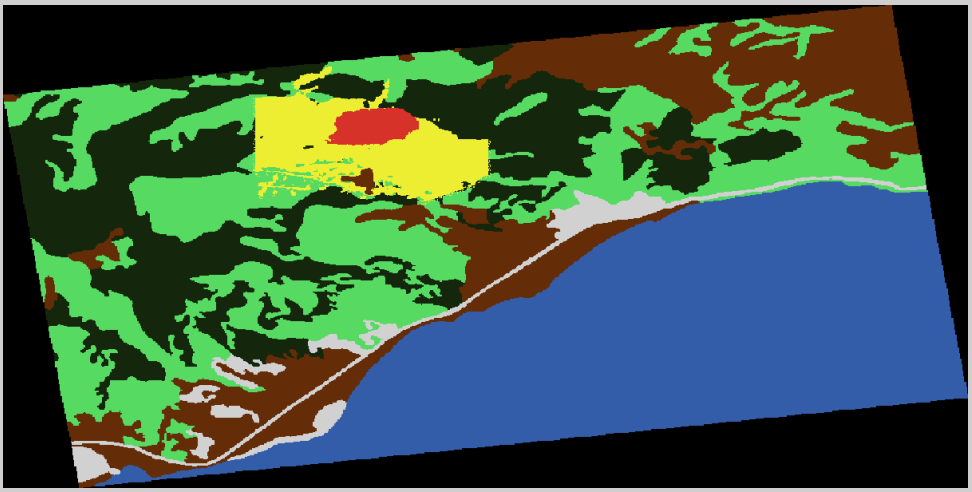




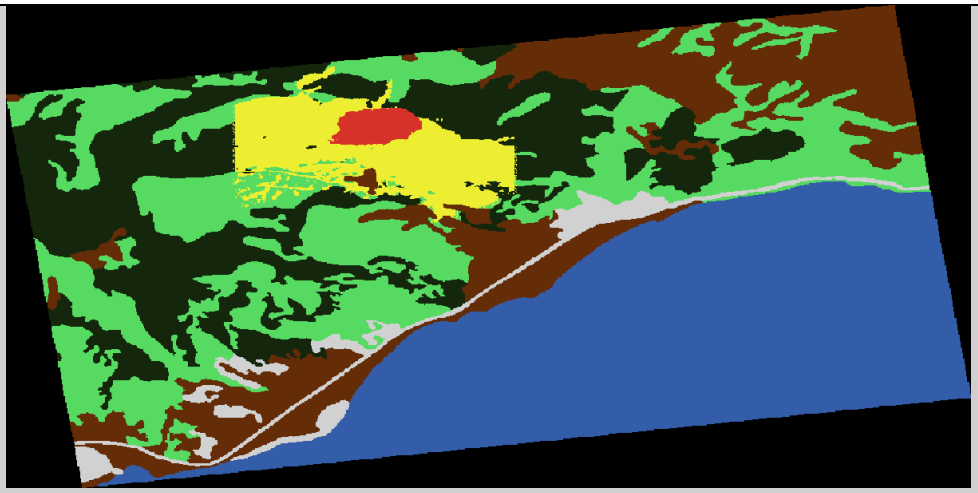
2



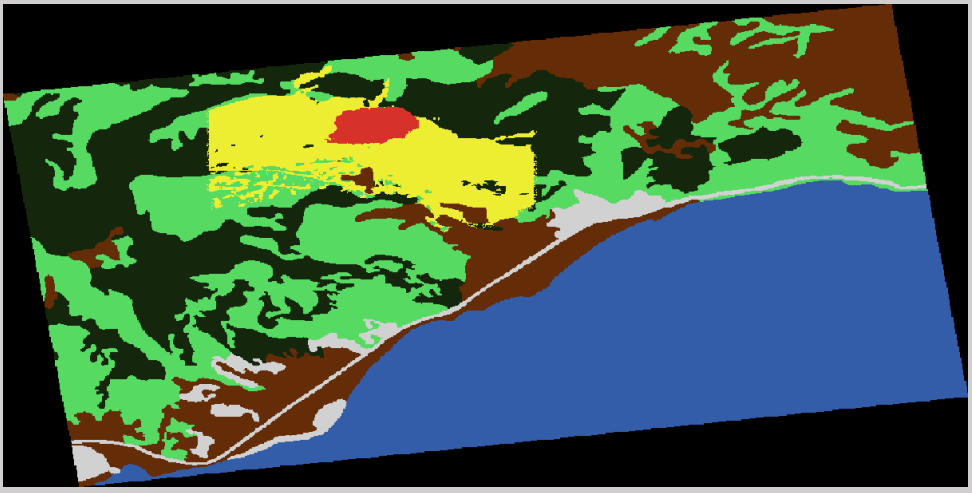
4



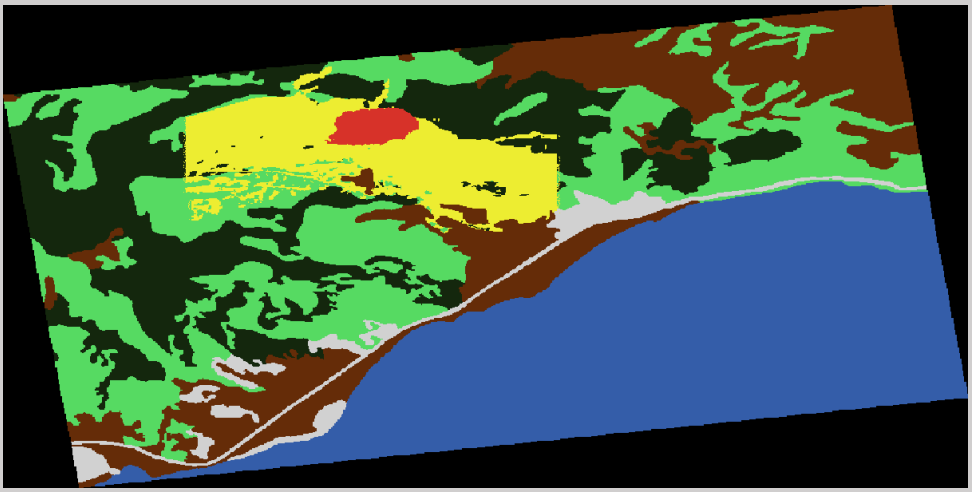
6



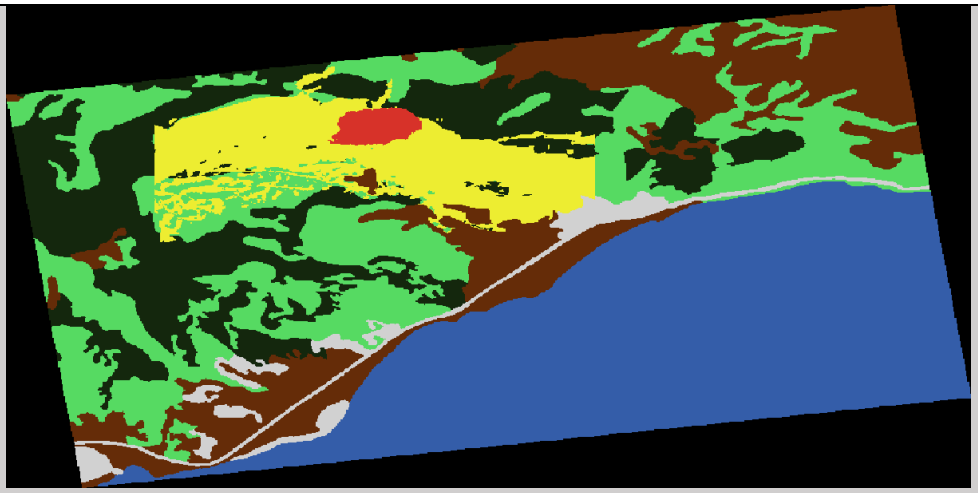
8



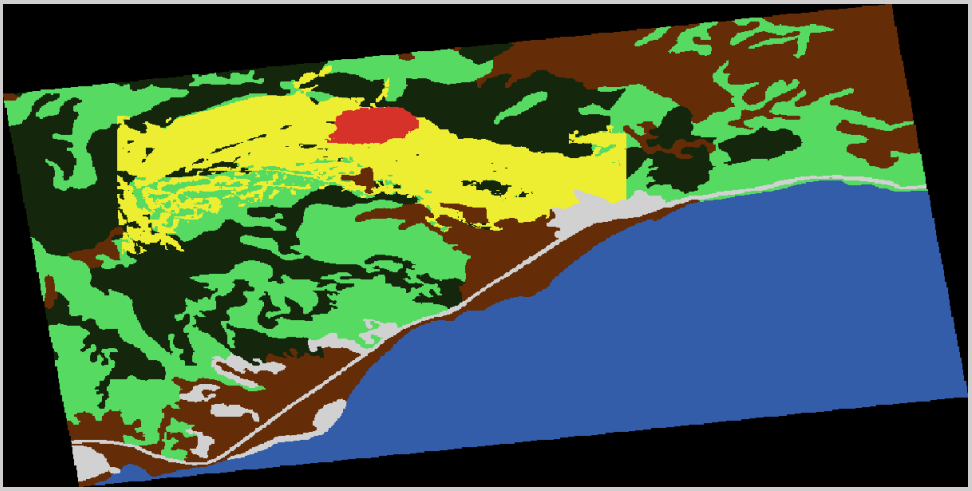
10



12



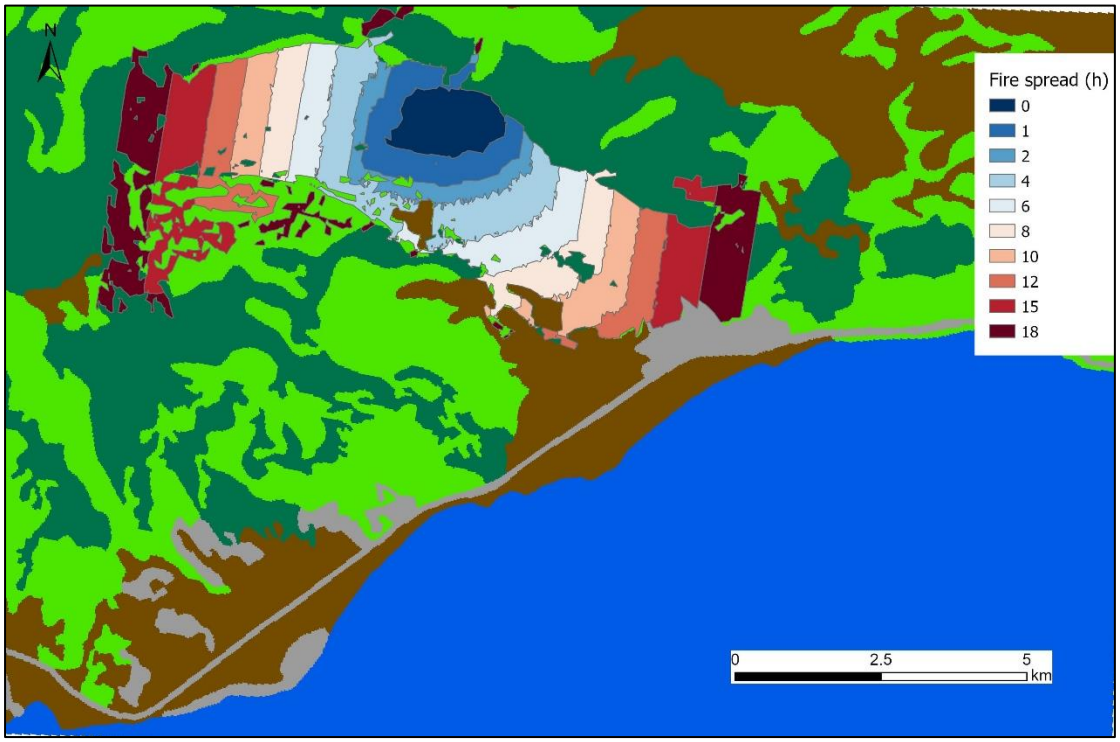
15



18

The figure above shows the result of the simulation every 2 hours except the last two which are every three hours. The fire spread can be seen expanding with the yellow color. The simulation outcome indicated that the fire spread is directing South, SouthEast and West. Moreover, the fire spread reaches the city of Kineta after 15 hours while it burns high forest density forest on the West direction. In addition, during the path of the fire spread there are some vacant areas that fire didn't burn, this is because the transition rules didn't apply for the selected cells. Finally, it can be said that the fire spreads faster in the first hours while slowing down when the model ends. This is because as the fire expands the area of the fire is becoming larger which has as a result to take more time for the model to calculate the transition rules. The figure below sums up the spread of the fire in the above hours as well as the area in which each hour takes place. The figure shows sharp lines at the end of each interval. This is because, the intermediate hours are not depicted in the figures, as result to show sharp lines. The next step is to identify how well the model simulated the real spread of the fire in Kineta.

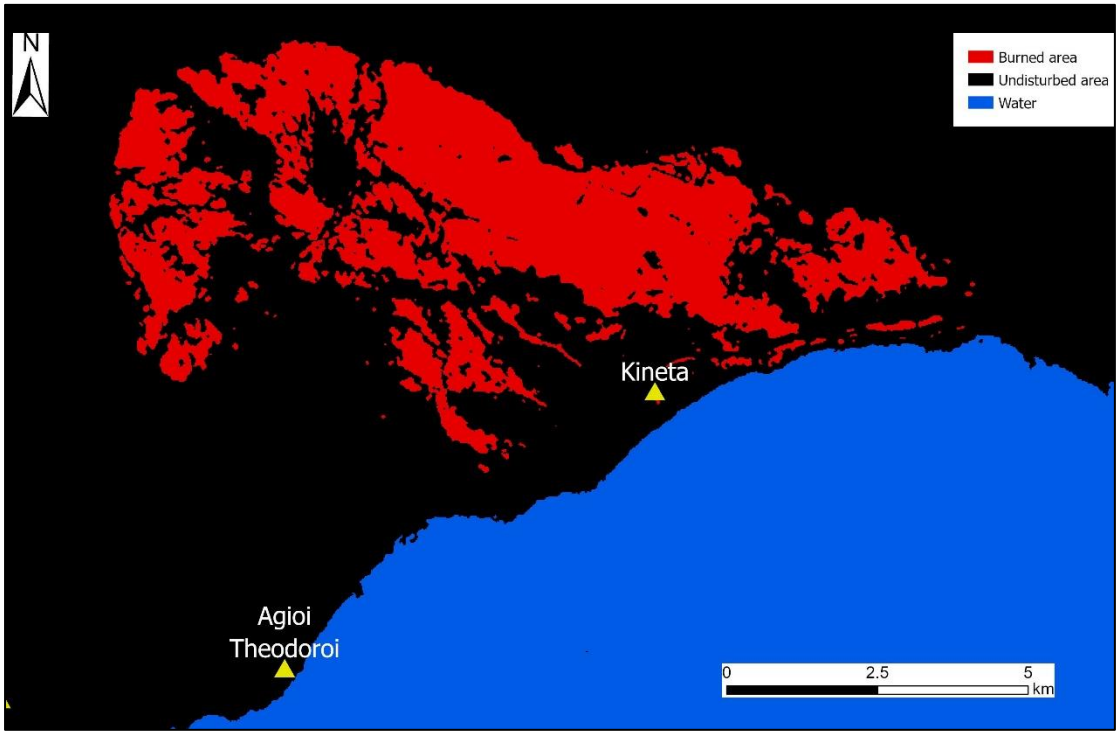
Figure 27: Area of fire spread every hour.



4.3 Validation test

The simulated model that performed in the previous chapter is based on the transition rules that was posed in the methodology chapter. This chapter examines how well the model simulated the actual fire spread. First, in order to extract the burned area of the actual fire spread the SVM classification method was selected. The results of the classification can be seen in the figure below.

Figure 28:Actual burned area in Kineta



The B12, B11, and B4 were selected to classify the model, because the images that were inserted in the inception V3 model have the same band combination. Furthermore, the accuracy of the classification was conducted by the tool *create accuracy assessment points* in ArcGis by creating 500 random points and then assigned the ground truth values to evaluate the accuracy of the classification. After that, the *confusion matrix* tool was selected to visualize the accuracy results, as can be seen in the figure below.

Table 7: Confusion Matrix results

ClassValue	Burned area	Undisturbed area	Water	Total	U_Accuracy	Kappa
Burned area	59	0	0	59	1	0
Undisturbed area	11	268	8	287	0.93	0
Water	0	0	154	154	1	0
Total	70	268	162	500	0	0
P_Accuracy	0.84	1	0.95	0	0.96	0
Kappa	0	0	0	0	0	0.93

The producer accuracy (P_Accurancy) or error of omission shows how well the classification results depict the actual burned area based on the sentinel-2 image, whereas the user accuracy (Accuracy) or errors of commission illustrate the incorrectly classified pixels. The results showed that 84% of the burned area was classified correctly, 100% of the undisturbed area we classified correctly and 95% of the water was classified correctly. On the other hand, the errors of commission are in all classes above 93% which means that the classified pixels that should belong in different classes is very low. Finally, the kappa statistic shows the overall assessment of the classification. The results indicate 93% which means that results of the classification have strong agreement with the actual data.

The classification indicate that the fire reaches the city of Kineta as the model simulated. The total burned area of the actual fire spread is 2979.59 hectares while the simulated the overlapping burned area with that of the simulation is 1766.26 hectares. This was done by using the *summarize within* tool in ArcGIS. Furthermore, a confusion matrix was created in order to validate the simulation with the SVM classification that was described above. For that reason, 500 random points were created. It is worth mentioning that, as ground truth we consider the SVM classification model that described above. The results of the confusion are depicted in the table below.

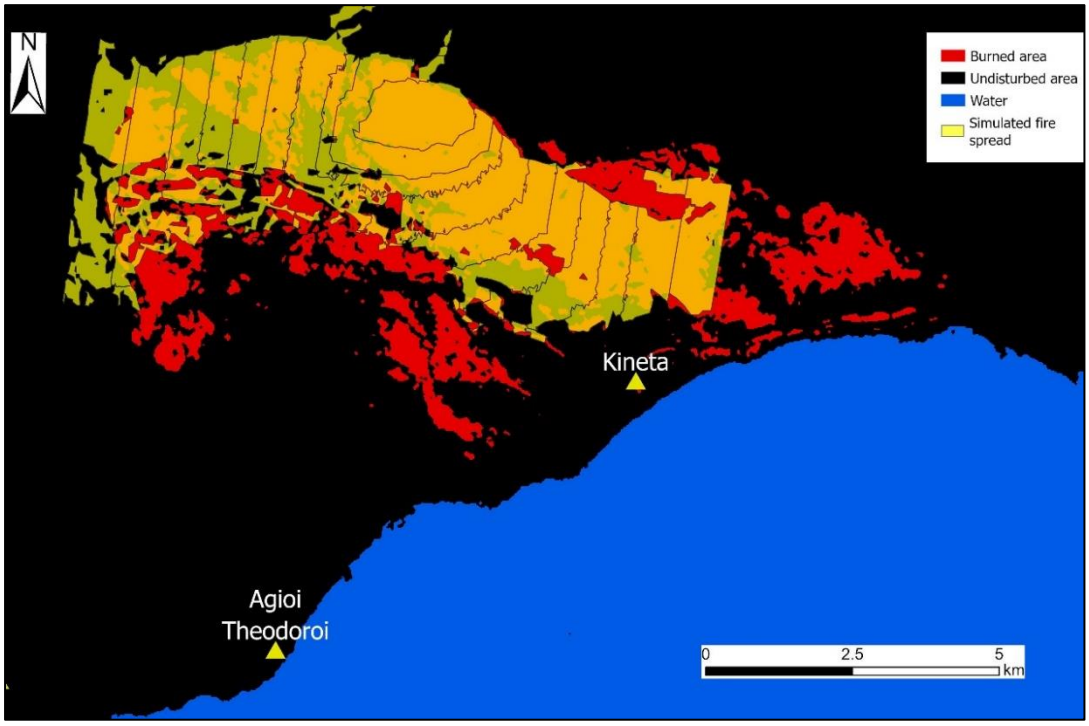
Table 8: Confusion Matrix results of the simulation model

ClassValue	Burned area	Non burned area	Total	U_Accuracy	Kappa
Burned area	41	18	59	0.69	0
Non burned area	22	419	441	0.95	0
Total	63	437	500	0	0
P_Accuracy	0.65	0.95	0	0.92	0
Kappa	0	0	0	0	0.62

As can be seen in the table above, the error of omissions is 65% in the burned area and 95% in the non-burned area, which signifies that 65% of the burned area is simulated correctly and 95% of the non-burned area is also simulated correctly. Moreover, the error of commissions is 69% in the burned area value and 95% in the non-burned area value which means that the simulation burned area pixels that should belong to non-burned area are not that many, whereas the non-burned area pixels that should belong to burned area are very low. Finally, the kappa statistics is 62% which means that the simulation model have moder-

ate agreement with the SVM classification model. The figure below depicts the burned and simulated fire spread areas.

Figure 29: Actual burned and simulated fire area.



As can be seen from the figure above a large area of the actual burned area is simulated by the model. The actual burned area expands more and reaches to the sea while the fire spread model stops when reaching to the city of Kineta. This means that the simulated model would probably have reached the sea if it had continued for a few hours. Furthermore, the simulated model expands even more on the West of the area. A possible explanation of this could be that the firefighters stopped the spread of the fire. Overall, the model simulated relatively well the actual fire spread based on the transition rules that was posed in the beginning of the model.

5. Conclusion

The main objective of this thesis was to work on a CNN model for wildfire outbreak detection using Sentinel-2 images and simulate forest fire spread based on several transition rules. Forest fires have caused serious problems in forest ecosystems in recent years, and the high frequency of forest fire outbreaks each year limits the possibility of physical recovery of forests. Greece, just like every other country with a Mediterranean climate, also faces thousands of forests fires every year. Therefore, the thesis focused on the area of Kineta in western Attica in Greece, where in 2018, the area faced a big wildfire that destroyed thousands of hectares of forest and human houses.

The applied methodology is separated into two parts: the fire prediction, which checks sentinel-2 images to see if there is fire or not, and the simulation of the fire spread, which imitates the route of the fire based on several transition rules. Specifically, images were collected in order to train the model by using the bands B12, B11, and B4 of the Sentinel-2 satellite. After that, the images were separated into training and testing and added to the Inception V3 model. The model showed high accuracy on random images, managing to achieve over 90% and 72% in f1-score, given the fact that the images for training the model were not very large. Moreover, the model can also predict larger images, however, the fire region does not depict them accurately.

The second part of this thesis was the forest fire simulation. For this part, data from the ESA Copernicus and Meteo satellites was extracted to identify the land cover type, the digital elevation model of the area, and the average wind speed. After that, the effects of the wind and the terrain were calculated in order to formulate the transition rules of the model. The results showed that the transition rules that were posed at the beginning of the model are relatively well simulated actual fire spread, managing to achieve 62% accuracy in the kappa statistics. The effect of the wind and the terrain are the most important factors that influence the spread of the forest fire.

Taking everything into consideration, this thesis answered the research questions that were posed at the beginning as well as the main objective. The proposed methodology was proven to be sufficiently robust to predict fire outbreaks by using Sentinel-2 images and relatively well to simulate forest fire spread based on the aforenoted transition rules. The inception V3 model presented satisfactory results despite the low availability of data. Therefore, the obtained results demonstrate the high potential of the implemented approach for an accurate model that can predict forest fire outbreaks. Moreover, the cellular automata model produced relatively accurate results with the defined transition rules. The proposed approach provides new insight into integrated protection for forest fire in different areas around the globe. Furthermore, the combination of machine learning with cellular automata and geographic information systems (GIS) can provide a useful tool to fight forest fires.

6. Discussion

As mentioned in the introduction section this thesis objective was to detect fire outbreaks by working with convolutional neural network and simulate fire spread using sentinel-2 images. For that reason, several sub questions have been posed in the beginning of this research. This section answers each question separately.

The first sub-question was to evaluate "how accurate this model can detect fire outbreaks." As can be seen in Chapter 4, the inception V3 model can detect fire outbreaks very accurately (over 90%) without collecting vast amounts of data. The second sub-question sought to ascertain "how the model performs on larger images." The results have shown that the model is able to detect if the image has fire with some accuracy, even though the fire region didn't depict it very accurately. These sub questions are answering the main research question, which was to identify "to what extent can a convolutional neural network detect forest fires?"

The second research question was to evaluate "to what extent can cellular automata simulate forest fire spread?" This research question also had two subquestions. The first sub-question is about the simulation of a fire spread model using cellular automata. This question aimed to identify "the factors that affect the route of a wildfire in a forest area." The main factors that this research showed are the effect of the wind and the effect of the terrain. These two factors are highly influential on the route of the fire's spread in our given scenario. Moreover, the type and density of the environment are also important factors that affect how the fire moves in a forest. The final sub-question was, "How accurately can this model simulate the spread of a wildfire by comparing it to previous fire outbreaks?" This research used the area of Kineta in Greece as an example due to the high destruction suffered in 2018. The results showed that the model simulated 62% based on the kappa statistic of the actual burned area, even though the actual extinguishing of the forest fire is unknown.

The connection between machine learning and cellular automata in forest fire detection and simulation is something that was missing from the literature review. This research was able to connect these two meanings and establish an effective framework in order to merge these concepts. However, this connection was done in very early stages, but it gives the possibility for further investigation.

A possible recommendation for further research is to use an image classification model like U-net that classifies the fire region more accurately. Moreover, as can be seen in the figure 23 the pixel-based classification captures clouds as a fire region this is because the classification is based on the pixel value. The process of image classification was not conducted in this research due to the high amount of training samples and data that is needed to build this model. However, it is believed that the image classification model would probably solve these problems. Furthermore, the ability to automatically classify the land cover type of any image and insert it to the cellular automata model is also a possibly topic for further research. Finally, there are plenty of image recognition machine learning models that exist today, the investigation and comparison with other models could be very useful information to identify which model performs better on the detection of fire outbreaks.

Bibliography

Albert, A., & Gonz, M. C. (n.d.). *Using Convolutional Networks and Satellite Imagery to Identify Pa erns in Urban Environments at a Large Scale*.

Alexandridis, A., Vakalis, D., Siettos, C. I., & Bafas, G. V. (2008). A cellular automata model for forest fire spread prediction: The case of the wildfire that swept through Spetses Island in 1990. *Applied Mathematics and Computation*, *204*(1), 191–201. https://doi.org/10.1016/j.amc.2008.06.046

Almeida, R. M., & MacAu, E. E. N. (2011). Stochastic cellular automata model for wildland fire spread dynamics. *Journal of Physics: Conference Series*, *285*(1). https://doi.org/10.1088/1742-6596/285/1/012038

Bodrožić, L., Stipaničev, D., & Šerić, M. (2006). Forest fires spread modeling using cellular automata approach. *… of Split, Department for Modelling and …*.

Byari, M., Bernoussi, A., Jellouli, O., Ouardouz, M., & Amharref, M. (2022). Multi-scale 3D cellular automata modeling: Application to wildland fire spread. *Chaos, Solitons and Fractals*, *164*(September), 112653. https://doi.org/10.1016/j.chaos.2022.112653

Cao, J., Yan, M., Jia, Y., Tian, X., & Zhang, Z. (2021). Application of a modified Inception-v3 model in the dynasty-based classification of ancient murals. *Eurasip Journal on Advances in Signal Processing*, *2021*(1). https://doi.org/10.1186/s13634-021-00740-8

Charan, S., Khan, M. J., & Khurshid, K. (2018). *Breast Cancer Detection in Mammograms using Convolutional Neural Network*. 0–4.

Chen, Y., Zhang, Y., Xin, J., Wang, G., Mu, L., Yi, Y., Liu, H., & Liu, D. (2019). UAV image-based forest fire detection approach using convolutional neural network. *Proceedings of the 14th IEEE Conference on Industrial Electronics and Applications, ICIEA 2019*, 2118–2123. https://doi.org/10.1109/ICIEA.2019.8833958

Crooks, A. (2017). Cellular automata. *Encyclopedia of Ecology*, *March*, 517–525. https://doi.org/10.1016/B978-0-444-63768-0.00147-5

de Almeida Pereira, G. H., Fusioka, A. M., Nassu, B. T., & Minetto, R. (2021). Active fire detection in Landsat-8 imagery: A large-scale dataset and a deep-learning study. *ISPRS Journal of Photogrammetry and Remote Sensing*, *178*, 171–186. https://doi.org/10.1016/j.isprsjprs.2021.06.002

Dewdney, A. K. (2018). Cellular automata. *Encyclopedia of Ecology*, 517–525. https://doi.org/10.1016/B978-0-444-63768-0.00147-5

Dong, N., Zhao, L., Wu, C. H., & Chang, J. F. (2020). Inception v3 based cervical cell classification combined with artificially extracted features. *Applied Soft Computing*, *93*, 106311. https://doi.org/10.1016/j.asoc.2020.106311

Duan, L., Ren, Y., & Duan, F. (2022). Adaptive stochastic resonance based convolutional neural network for image classification. *Chaos, Solitons and Fractals*, *162*, 112429. https://doi.org/10.1016/j.chaos.2022.112429

Georgiev, G. D., Hristov, G., Zahariev, P., & Kinaneva, D. (2020). Forest Monitoring System for Early Fire Detection Based on Convolutional Neural Network and UAV imagery. *28th National Conference with International Participation, TELECOM 2020 - Proceedings*, *9*(17 2), 57–60. https://doi.org/10.1109/TELECOM50385.2020.9299566

Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J., & Chen, T. (2018). Recent advances in convolutional neural networks. *Pattern Recognition*, *77*, 354–377. https://doi.org/10.1016/j.patcog.2017.10.013

Handa, N., Kaushik, Y., Sharma, N., Dixit, M., & Garg, M. (2021). Image Classification Using Convolutional Neural Networks. *Communications in Computer and Information Science*, *1393*, 510–517. https://doi.org/10.1007/978-981-16-3660-8_48

Hernández Encinas, L., Hoya White, S., Martín del Rey, A., & Rodríguez Sánchez, G. (2007). Modelling forest fire spread using hexagonal cellular automata. *Applied Mathematical Modelling*, *31*(6), 1213–1227. https://doi.org/10.1016/j.apm.2006.04.001

Hong, Z., Tang, Z., Pan, H., Zhang, Y., Zheng, Z., Zhou, R., Ma, Z., Zhang, Y., Han, Y., Wang, J., & Yang, S. (2022). Active Fire Detection Using a Novel Convolutional Neural Network Based on Himawari-8 Satellite Images. *Frontiers in Environmental Science*, *10*(March), 1–11. https://doi.org/10.3389/fenvs.2022.794028

Hu, X., Ban, Y., & Nascetti, A. (2021a). Sentinel-2 MSI data for active fire detection in major fire-prone biomes: A multi-criteria approach. *International Journal of Applied Earth Observation and Geoinformation*, *101*(November), 102347. https://doi.org/10.1016/j.jag.2021.102347

Hu, X., Ban, Y., & Nascetti, A. (2021b). Sentinel-2 MSI data for active fire detection in major fire-prone biomes: A multi-criteria approach. *International Journal of Applied Earth Observation and Geoinformation*, *101*, 102347. https://doi.org/10.1016/j.jag.2021.102347

Hubel, D. H., & Wiesel, T. N. (1968). Receptive fields and functional architecture of monkey striate cortex. *The Journal of Physiology*, *195*(1), 215–243. https://doi.org/https://doi.org/10.1113/jphysiol.1968.sp008455

Jahandad, Sam, S. M., Kamardin, K., Amir Sjarif, N. N., & Mohamed, N. (2019). Offline signature verification using deep learning convolutional Neural network (CNN) architectures GoogLeNet inception-v1 and inception-v3. *Procedia Computer Science*, *161*, 475–483. https://doi.org/10.1016/j.procs.2019.11.147

Lecun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). A B7CEDGF HIB7PRQTSUDGQICWVYX HIB edCdSISIXvg5r ` CdQTw XvefCdS. *Proc. OF THE IEEE*. http://ieeexplore.ieee.org/document/726791/#full-text-section

Lin, M., Chen, Q., & Yan, S. (2014). Network in network. *2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings*, 1–10.

Lopez Pinaya, W. H., Vieira, S., Garcia-Dias, R., & Mechelli, A. (2019). Convolutional neural networks. In *Machine Learning: Methods and Applications to Brain Disorders*. Elsevier Inc. https://doi.org/10.1016/B978-0-12-815739-8.00010-9

Mazzeo, G., De Santis, F., Falconieri, A., Filizzola, C., Lacava, T., Lanorte, A., Marchese, F., Nolè, G., Pergola, N., Pietrapertosa, C., & Satriano, V. (2022). Integrated Satellite System for Fire Detection and Prioritization. *Remote Sensing*, *14*(2), 1–25. https://doi.org/10.3390/rs14020335

Neumann, J. von. (1966). *Theory-of-self-reproducing-automata.pdf* (p. 403).

O'Shea, K., & Nash, R. (2015). *An Introduction to Convolutional Neural Networks*. 1–11. http://arxiv.org/abs/1511.08458

Priya, R. S., & Vani, K. (2019). Deep learning-based forest fire classification and detection in

satellite images. *Proceedings of the 11th International Conference on Advanced Computing, ICoAC 2019*, 61–65. https://doi.org/10.1109/ICoAC48765.2019.246817

Quartieri, J., Mastorakis, N. E., Iannone, G., & Guarnaccia, C. (2010). A Cellular Automata model for fire spreading prediction. *International Conference on Urban Planning and Transportation - Proceedings*, *May 2014*, 173–179.

Rainey, K., Reeder, J. D., & Corelli, A. G. (2016). Convolution neural networks for ship type recognition. *Automatic Target Recognition XXVI*, *9844*(November), 984409. https://doi.org/10.1117/12.2229366

Reimers, C., & Requena-Mesa, C. (2020). Deep Learning - an Opportunity and a Challenge for Geo- and Astrophysics. In *Knowledge Discovery in Big Data from Astronomy and Earth Observation: Astrogeoinformatics*. Elsevier Inc. https://doi.org/10.1016/B978-0-12-819154-5.00024-2

Rostami, A., Shah-Hosseini, R., Asgari, S., Zarei, A., Aghdami-Nia, M., & Homayouni, S. (2022). Active Fire Detection from Landsat-8 Imagery Using Deep Multiple Kernel Learning. *Remote Sensing*, *14*(4). https://doi.org/10.3390/rs14040992

Safi, Y., & Bouroumi, A. (2013). Prediction of forest fires using artificial neural networks. *Applied Mathematical Sciences*, *7*(5–8), 271–286. https://doi.org/10.12988/ams.2013.13025

Szegedy, C., Liu, W., & Yangqing, J. (2015). Measurement Reliability and Validity. *Research Methods in Applied Settings*, 319–338. https://doi.org/10.4324/9781410605337-29

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the Inception Architecture for Computer Vision. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *2016-Decem*, 2818–2826. https://doi.org/10.1109/CVPR.2016.308

Wang, C., Chen, D., Hao, L., Liu, X., Zeng, Y., Chen, J., & Zhang, G. (2019). Pulmonary image classification based on inception-v3 transfer learning model. *IEEE Access*, *7*, 146533–146541. https://doi.org/10.1109/ACCESS.2019.2946000

Wang, X., Chang, L., Liu, J., Qin, X., Ning, W., & Zhou, W. (2017). A cellular automata model for forest fire spreading simulation. *2016 IEEE Symposium Series on Computational Intelligence, SSCI 2016*, *71533001*. https://doi.org/10.1109/SSCI.2016.7849971

Wu, J. (2017). Introduction to Convolutional Neural Networks. *Introduction to Convolutional Neural Networks*, 1–31. https://web.archive.org/web/20180928011532/https://cs.nju.edu.cn/wujx/teaching/15_CNN.pdf

Wu, X., Lin, A. Q., Li, Y., Wu, H., Cen, L. Y., Liu, H., & Song, D. X. (2021). Simulating spatiotemporal land use change in middle and high latitude regions using multiscale fusion and cellular automata: The case of Northeast China. *Ecological Indicators*, *133*, 108449. https://doi.org/10.1016/j.ecolind.2021.108449

Wu, Z., Li, M., Wang, B., Quan, Y., & Liu, J. (2021). Using artificial intelligence to estimate the probability of forest fires in heilongjiang, Northeast China. *Remote Sensing*, *13*(9). https://doi.org/10.3390/rs13091813

Wu, Z., Wang, B., Li, M., Tian, Y., Quan, Y., & Liu, J. (2022). Simulation of forest fire spread based on artificial intelligence. *Ecological Indicators*, *136*(December 2021), 108653. https://doi.org/10.1016/j.ecolind.2022.108653

Xu, Y., & Zhang, H. (2022). Convergence of deep convolutional neural networks. *Neural Networks*, *153*, 553–563. https://doi.org/10.1016/j.neunet.2022.06.031

Yang, J., Su, J., Chen, F., Xie, P., & Ge, Q. (2016). A local land use competition cellular automata model and its application. *ISPRS International Journal of Geo-Information*, *5*(7). https://doi.org/10.3390/ijgi5070106

Yeturu, K. (2020). Machine learning algorithms, applications, and practices in data science. In *Handbook of Statistics* (1st ed., Vol. 43). Elsevier B.V. https://doi.org/10.1016/bs.host.2020.01.002

Giabbanelli, P. J. (2019). Visual Analytics to Identify Temporal Patterns and. 29(1), 1–26.

Alexandridis, A., Vakalis, D., Siettos, C. I., & Bafas, G. V. (2008). A cellular automata model for forest fire spread prediction: The case of the wildfire that swept through Spetses Island in 1990. *Applied Mathematics and Computation*, *204*(1), 191–201. https://doi.org/10.1016/j.amc.2008.06.046

Angeles-camacho, C. (2021). *Methodologies Used in the Extrapolation of Wind Speed Data at Different Heights and Its Impact in the Wind Energy Resource Assessment in a Region*. *June 2011*. https://doi.org/10.5772/20669

Dewdney, A. K. (2018a). Cellular automata. *Encyclopedia of Ecology*, 517–525. https://doi.org/10.1016/B978-0-444-63768-0.00147-5

Dewdney, A. K. (2018b). Cellular automata. In *Encyclopedia of Ecology* (Second Edi, Vol. 2). Elsevier. https://doi.org/10.1016/B978-0-444-63768-0.00147-5

Erdik, T., & Law, P. (2012). *Wind Velocity Vertical Extrapolation by Extended Power Law*. *2012*. https://doi.org/10.1155/2012/178623

Freire, J. G., & Dacamara, C. C. (2018). *Using cellular automata to simulate wildfire propagation and to assist in fire prevention and fighting*. *November*, 1–17.

Giabbanelli, P. J. (2019). *Visual Analytics to Identify Temporal Patterns and*. *29*(1), 1–26.

Goutte, C., & Gaussier, E. (2005). A Probabilistic Interpretation of Precision, Recall and F-Score, with Implication for Evaluation. *Lecture Notes in Computer Science*, *3408*(April), 345–359. https://doi.org/10.1007/978-3-540-31865-1_25

Mutthulakshmi, K., Wee, M. R. E., Wong, Y. C. K., Lai, J. W., Koh, J. M., Acharya, U. R., & Cheong, K. H. (2020). Simulating forest fire spread and fire-fighting using cellular automata. *Chinese Journal of Physics*, *65*(November 2019), 642–650. https://doi.org/10.1016/j.cjph.2020.04.001

Noble, W. S. (2006). What is a support vector machine? *Nature Biotechnology*, *24*(12), 1565–1567. https://doi.org/10.1038/nbt1206-1565

Pintor, A., Pinto, C., Mendonça, J., Pilão, R., & Pinto, P. (2022). *Insights on the use of wind speed vertical extrapolation methods Insights on the use of wind speed vertical extrapolation methods*. *September*. https://doi.org/10.24084/repqj20.410

Ting, K. M. (2017). Confusion Matrix. *Encyclopedia of Machine Learning and Data Mining*, *October*, 260–260. https://doi.org/10.1007/978-1-4899-7687-1_50

Engineering toolbox 2020, Resources, Tools and Basic Information for Engineering and Design of Technical Applications accessed 02/02/2023, https://www.engineeringtoolbox.com/wind-shear-d_1215.html

Geemap 2022 accessed 20/11/2022, https://geemap.org/

Appendices

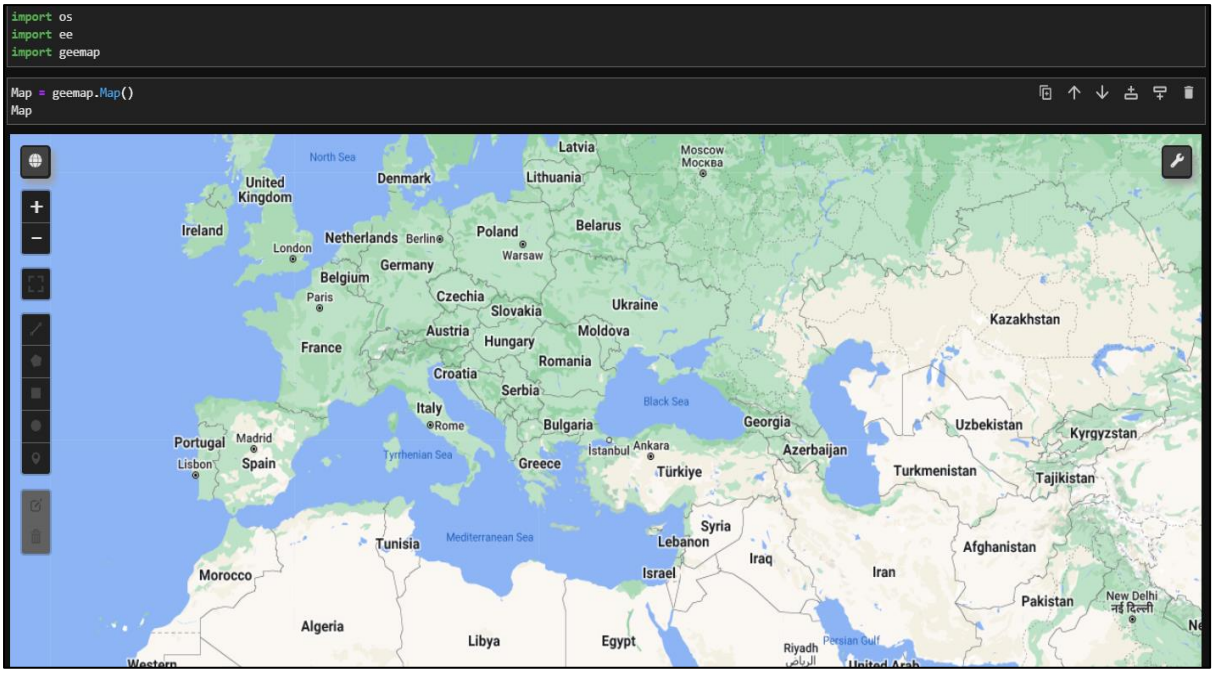
Appendix I – Land cover reclassification

Corine land cover-2018	New classes
-	Fire
• Coniferous forest • Mixed forest	High density forest
• Natural grasslands • Sclerophyllous vegetation • Transitional woodland-shrub	Low density forest
• Complex cultivation patterns • Non-irrigated arable land • Olive groves • Sparsely vegetated areas • Sport and leisure facilities • Vineyards	Barren soil
• Discontinuous urban fabric • Industrial or commercial units • Road and rail networks and associated land	Human structures
• Sea and ocean	Water

Appendix II: Steps for image collection

This section describes the process of collecting images in order to train and validate the inception V3 model. The collection of the images was carried out using the geemap package (geemap, 2022) and Google Earth Engine. The figure below depicts the display of geemap in the jupyter lab environment.

Figure 30: Map display using geemap package.



After that, the selection of the region we want to extract is carried out and by using ImageCollection of Google Earth Engine package the sentinel 2 satellite is selected using the B12, B11, B4 bands, as it mentioned in the previous chapter. Thenceforth, the selected visualize parameters are set in order to view the image. This can be seen in the figure below.

Figure 31: Image collection and visualization

```
region = Map.user_roi
```

```
region.getInfo()
```

```
...
```

```
collection  =  (ee.ImageCollection('COPERNICUS/S2_SR_HARMONIZED')
           .filterBounds(region)
           #.filter('CLOUDY_PIXEL_PERCENTAGE < 5')
           .filterDate('2018-07-15', '2018-07-22')
           .select('B12','B11','B4'))
```

```
color = ['FFFFFF', 'CE7E45', 'DF923D', 'F1B555', 'FCD163', '99B718',
              '74A901', '66A000', '529400', '3E8601', '207401', '056201',
              '004C00', '023B01', '012E01', '011D01', '011301']
pallete = {"min":0, "max":1, 'palette':color}
```

```
vis_params = {
    'min': 0,
    'max': 6000,
    'gamma': 1.4,
}

Map.addLayer(collection, vis_params, "sentinel")
Map.setCenter(21.706095, 40.199855, 12)
Map
```

Finally, the preferences file is selected and by using the geemap package the images are download in '*jpg'* format.

Figure 32: Extraction of images

```
out_dir = os.path.expanduser("~/Desktor")

geemap.get_image_collection_thumbnails(collection,out_dir, vis_params,region=region)
```

Appendix III: Steps of fire detection

This section describes the steps to develop the model for forest fire detection. Firstly, after separating the data into train and testing the ImageDataGenerator was used to upload the images into the model. After that, in order increase the accuracy of training even further the data augmentation technique was used (zooming and horizontal flipping). This can be better be seen in the figure below.

Figure 33: ImageDataGenerator

```
TRAINING_DIR = r"[redacted]\data\Train"
training_datagen = ImageDataGenerator(rescale=1./255,
zoom_range=0.15,
horizontal_flip=True,
fill_mode='nearest')

VALIDATION_DIR = r[redacted]\data\test"
validation= ImageDataGenerator(rescale = 1./255)

train_generator = training_datagen.flow_from_directory(
TRAINING_DIR,
shuffle = True,
class_mode='categorical',
batch_size = 20)

validation_generator = validation.flow_from_directory(
VALIDATION_DIR,
class_mode='categorical',
shuffle = True,
batch_size= 15)
```

After that the Inception-V3 model was imported using the Keras module. This research uses the Global Average Pooling which it has two important advantages over the fully connected layers. The first one is that the Global Average Pooling is more innate to the convolution design by establishing correlations between feature maps and categories and the second one is that the overfitting is prevented (Lin et al., 2014). In addition, 2 dropout layers and 2 dense layers was used in order to make sure that the model doesn't overfit. Finally, 2 classes were added using the SoftMax layer. This can be better be seen in the figure below.

Figure 34: Inception V3 model structure

```
base_model = InceptionV3(weights='imagenet', include_top=False)
x = base_model.output
x = GlobalAveragePooling2D()(x)
x = Dense(2048, activation='relu')(x)
x = Dropout(0.25)(x)
x = Dense(1024, activation='relu')(x)
x = Dropout(0.2)(x)
prediction = Dense(2, activation='softmax')(x)
model = Model(inputs=base_model.input, outputs=prediction)
for layer in base_model.layers:
    layer.trainable = False
model.compile(optimizer='rmsprop', loss='categorical_crossentropy', metrics=['acc'])
history = model.fit(
train_generator,
steps_per_epoch = 4,
epochs = 20,
validation_data = validation_generator,
validation_steps = 4)
```

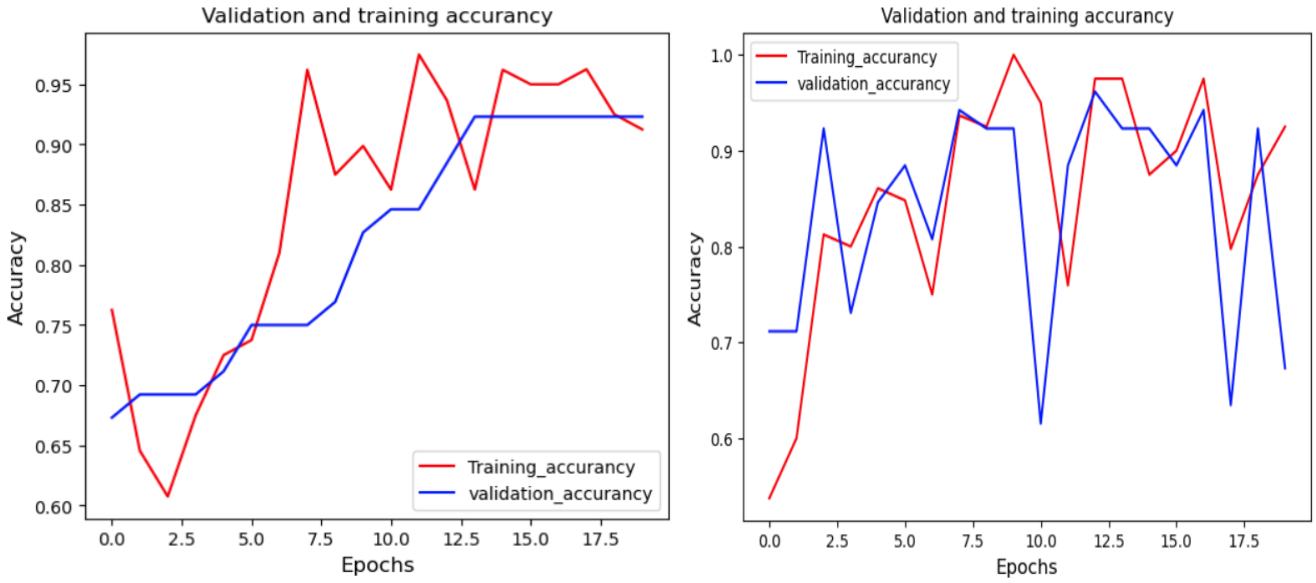
As can be seen in the figure above the layers was trained for 20 epochs. After that, the model freezes the 249 layers to inhibits the modification of its weights. The code can be seen in the following figure.

Figure 35: Freezing the layers

```
for layer in model.layers[:249]:
    layer.trainable = False
for layer in model.layers[249:]:
    layer.trainable = True
from tensorflow.keras.optimizers import SGD
model.compile(optimizer=SGD(lr=0.0001, momentum=0.9), loss='categorical_crossentropy', metrics=['acc'])
history = model.fit(
train_generator,
steps_per_epoch = 4,
epochs = 20,
validation_data = validation_generator,
validation_steps = 4)
```

According to the figure 16 the first run of the model the accuracy reaches almost 93% while the training accuracy reaches less than 70%. After freezing the first 249 layers the model seems to gain over 90% of validation accuracy while the train accuracy remains almost stable. These figures are emphasizing the importance of the freezing the model as it gained more validation accuracy and reduces the computation time. Moreover, after training the model, the validation process will take random images from a folder that it contains fire and non-fire images which were not included in the algorithm and test the accuracy of the results.

Figure 36: Validation and training accuracy after and before freezing the model. On the left is before and on the right is after.

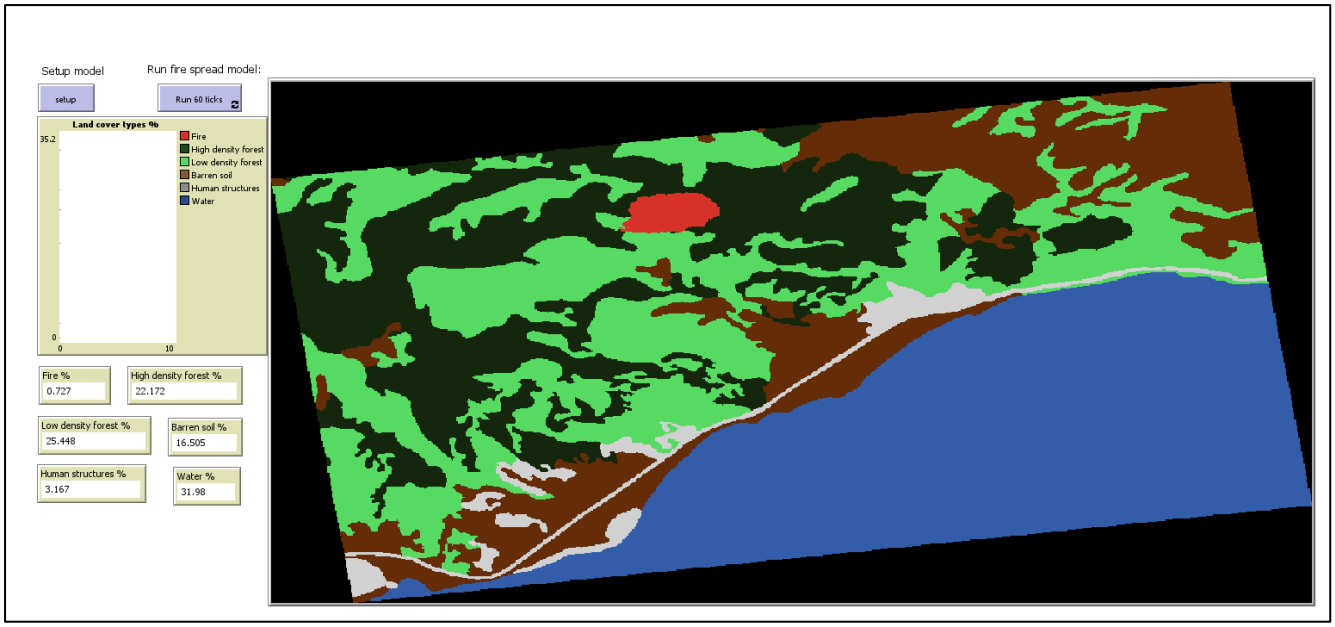


It is worth mentioning that, both the training and validation accuracy are considered to be optimal, given the fact that the data are not very large and for that reason inceptionV3 is considered better when working with small datasets.

Appendix IV: Description of simulation model

This section describes the NetLog model which the simulation of forest fire spread was carried out. Firstly, the interface of model is depicted in the figure below. The interface consists of the Setup and Run bottoms which they load and run the model, a graph that calculate the percentage of change in land cover and the percentage of land cover in each tick.

Figure 37: NetLogo interface



The code for the Setup and load the raster file's function can be seen in the figure below. The setup clears the model, and it loads the raster datasets and resets the ticks back to 0. The load-gis sets all the raster datasets and it uses as an interface the land cover map. The setup-patches sets the land cover types as well as the neighborhood which the given cell takes into account. Moreover, the update-display gives the land cover types the preferable color to visualize.

Figure 38: Setup procedure

```
to setup
  clear-all
  load-gis
  setup-patches
  update-display
  reset-ticks
  print "Finished"
end

to load-gis
  clear-all
  set land-cover-map gis:load-dataset "land_cover.asc"
  set slope-map gis:load-dataset "slope.asc"
  set aspect-map gis:load-dataset "aspect.asc"
  set fire-propagation-map gis:load-dataset "fire_propagation.asc"
  gis:set-world-envelope-ds gis:envelope-of land-cover-map
end
```

```
to setup-patches
  gis:apply-raster land-cover-map land-cover
  gis:apply-raster slope-map slope-suitability
  gis:apply-raster fire-propagation-map firepropa-suitability
  gis:apply-raster aspect-map aspect-suitability
  let landcovertypes ["Fire" "High density forest" "Low density forest" "Barren soil" "Human
structures" "Water"]
  let landcovernumbers [1 2 3 4 5 6]
  let landcover table:from-list (map list landcovernumbers landcovertypes)

  set nearby [[-1 1] [1 1] [-1 -1] [-1 0]……[-3 2][-3 1]]

  print "Map setup done."

  estimate_initial-%
end

to update-display
  print "Setting up patch colors"
  ask patches
  [
    if land-cover = 1 [ set pcolor 15 ]
    if land-cover = 2 [ set pcolor 51 ]
    if land-cover = 3 [ set pcolor 66 ]
    if land-cover = 4 [ set pcolor 22 ]
    if land-cover = 5 [ set pcolor 8 ]
    if land-cover = 6 [ set pcolor 105]
    ]
  print "Setting up patch colors done."
end
```

After that, the model calculates the percentage of each land cover type when the model sets up and displays it to the graph. The figure below is an example of calculation of fire.

Figure 39: Calculation of fire percentage

```
to estimate_initial-%
  if count patches with [land-cover = 1] != 0 [set %Fire (count patches with [land-cover = 1]) /
(count patches with [land-cover = 1 or land-cover = 2 or land-cover = 3 or land-cover = 4 or
land-cover = 5 or land-cover = 6]) * 100]
end
```

Following that, the figure below depicts the countneighbors_by_type function which calculates the total number of cell type for each land cover type in the defining area. This is a very important function because it has a critical role in simulating the fire spread model.

Figure 40: Counting of total neighbors.

```
to countneighbors_by_type
  ask patches
  [
   if land-cover != 1 [
   let count_fire 0
   let count_highdensityforest 0
   let count_lowdensityforest  0
   let count_barrensoil 0
   let count_humanstructure 0

   ask patches at-points nearby [

    if land-cover = 1 [
     set count_fire count_fire + 1
    ]
    if land-cover = 2 [
     set count_highdensityforest count_highdensityforest + 1
    ]
    if land-cover = 3 [
     set count_lowdensityforest count_lowdensityforest + 1
    ]
    if land-cover = 4 [
     set count_barrensoil count_barrensoil + 1
    ]
    if land-cover = 5 [
     set count_humanstructure count_humanstructure + 1
    ]
   ]
  set neighborcount_fire count_fire
  set neighborcount_highdensityforest count_highdensityforest
  set neighborcount_lowdensityforest count_lowdensityforest
  set neighborcount_barrensoil count_barrensoil
  set neighborcount_humanstructure count_humanstructure
  ]
  ]
end
```

Consequently, the fire-birth function assigns the new color when the transition rules are applicable, which means that new fire has been created.

Figure 41: Fire birth funtion

```
to fire-birth
 set pcolor 45
 set land-cover 1
end
```

Finally, the code that run all the above as well as the transition rules is depicted in the figure below. The models run for 72 ticks and then it stops, and the outcome of each tick can be seen in the interface of the model.

Figure 42: Fire spread code

```
to go-firespreadmodel

 if ticks >= 72 [ stop ]

 print "Fire spread model"

 let fireexpa 0
 let oc1-1 0
 let oc1-2 0
 let oc1-3 0
 let oc1-4 0

 print "Counting neighborhoods and counting types within the 75x75m neighborhood ..."
 countneighbors_by_type

 print "Start of transition rules."
 ask patches [

  let slope 0
  let aspect 0
  set aspect aspect-suitability
  set slope slope-suitability
  let firepro 0
  set firepro firepropa-suitability

  if (land-cover != 6) and (land-cover != 4) and (neighborcount_Fire > 0) and (land-cover !=
5) [
    let probabilityhighdensity (0.58 * (1 + 0.4) * 1.03 * slope * firepro)
    let probabilitylowdensity (0.58 * (1 + 0.2) * (1 + 0.2) * slope * firepro)
    if  (probabilityhighdensity > 0.018) and (neighborcount_highdensityforest > neighbor-
count_lowdensityforest) and (aspect >= 67.5 and aspect <= 292.5) [
      fire-birth
      set fireexpa fireexpa + 1
      print "fire due to outcome 1"
      set oc1-1 oc1-1 + 1]
    if (probabilitylowdensity > 0.013) and (neighborcount_lowdensityforest > neighbor-
count_highdensityforest) and (aspect >= 67.5 and aspect <= 292.5) [
      fire-birth
      set fireexpa fireexpa + 1
      print "fire due to outcome 2"
      set oc1-1 oc1-1 + 1]
      set oc1-1 oc1-1 + 1]]]
 print "Transition rules done"
 print word "Resulting new fire cells in this tick:" fireexpa
 print "Outcomes:"
 print word "Total new cells due to Outcome 1: " oc1-1
```

```
  print word "Total new cells due to Outcome 2: " oc1-2
  print word "Total new cells due to Outcome 3: " oc1-3
  print word "Total new cells due to Outcome 4: " oc1-4

  estimate_initial-%
  print "Simulation finished."
  tick
end
```