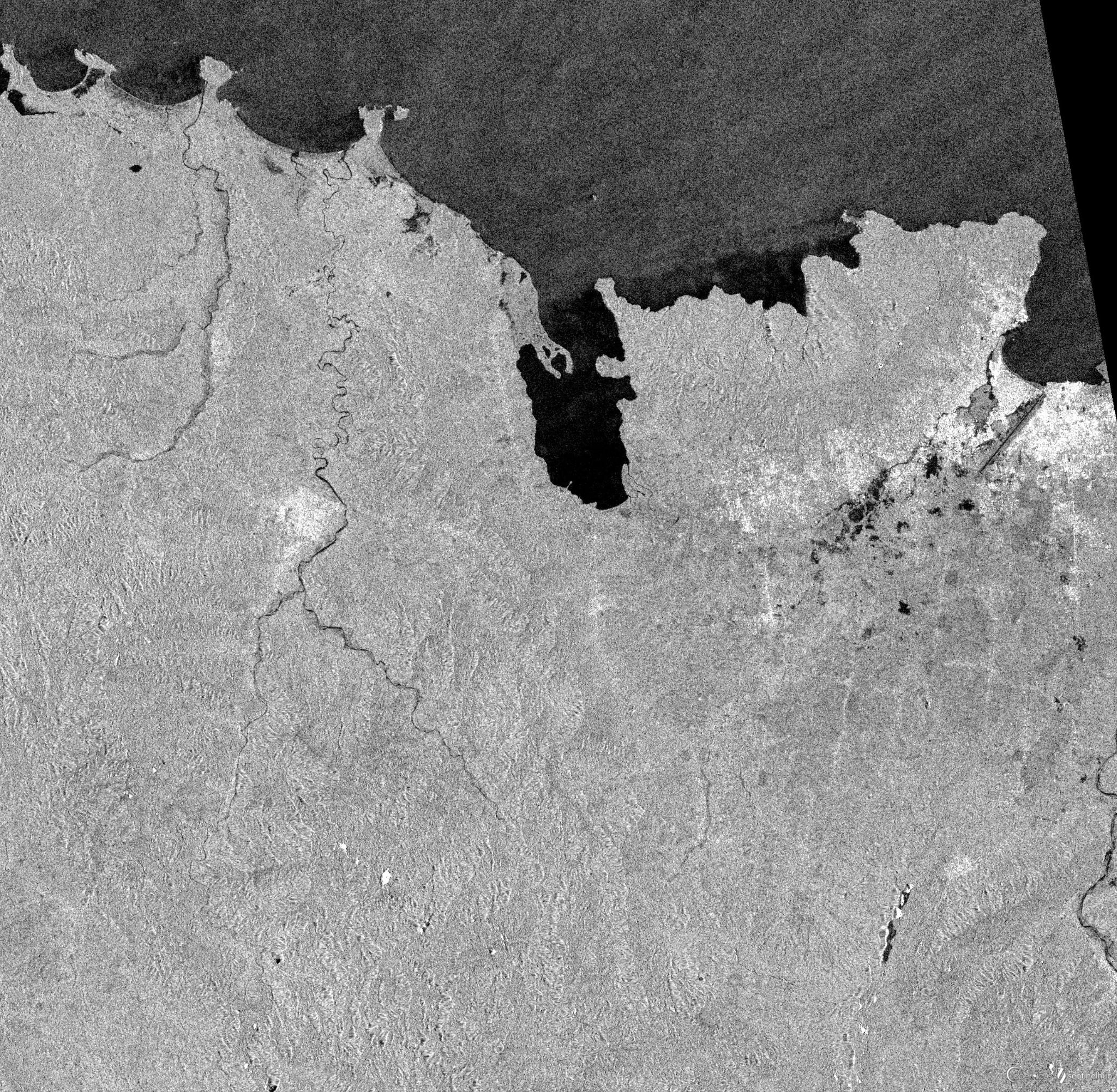
Flood risk assessment using time series analysis

June 2022



**Authors**: Jakko-Jan van Ek, Julia Sipkema, Mark Boeve, Niels Verouden & Raimon Bach Pareja

**Commissioner**: S.M.J. Arts, Ministerie van Defensie

**Coach**: Sytze de Bruin

Table of contents

[**Executive summary**](#_rs5gk8uop33g) **3**

[**1. Introduction**](#_sc6e2th1xdqb) **4**

[1.1 Background](#_37ahxala70ma) 4

[1.2 Problem analysis](#_5n3xbtk4ip0d) 4

[1.3 Objective](#_295batsrskmt) 5

[**2. Materials**](#_8m6nhzkm4e8z) **6**

[2.1 Synthetic Aperture Radar (SAR)](#_47j32uw1snjk) 6

[2.2 Digital Elevation Model (DEM)](#_ci5xrhz84d2m) 7

[2.3 Water bodies](#_yl5069wh3tyh) 7

[2.4 Global Human Settlement](#_6w59yjk6zwb8) 7

[2.5 Precipitation](#_po24w2a4gy) 7

[**3. Methods**](#_wwmbwruqfj53) **8**

[3.1 Data acquisition](#_ei6uf6larnar) 8

[3.1.1 Sentinel Hub EO Browser](#_xjmcjglxoxh7) 9

[3.1.2 EO Learn API](#_oqg80e1g6ot9) 9

[3.2 Supervised Classification with Machine Learning](#_1jr50lgcrvid) 9

[3.2.1 Training Data and Processing Steps](#_4ikj7pnrd9y4) 9

[3.2.2 Grid Search and Cross Validation](#_7vuqjr1pdjvh) 10

[3.2.3 Gaussian Naive Bayes Classifier (GNB)](#_3l704p2b1ebi) 10

[3.2.4 K-nearest neighbours (kNN)](#_4p00e0bto2c1) 10

[3.2.5 Support vector machines (SVM)](#_rd3gcuoechgq) 10

[3.2.6 Random Forest (RF)](#_9gzxtkcdkmye) 10

[3.2.7 Predictions and Flood Frequency Analysis](#_f5n9vp18e6qx) 11

[3.3 Histogram thresholding](#_2cx6t6wsvuq5) 11

[3.4 Image differencing of SAR imagery](#_pf08vxhg1pvz) 11

[3.5 Urban flood detection with time series analysis](#_ywcto29mevq0) 12

[3.5.2 Data preprocessing and loading](#_2lz5vubqtg46) 12

[3.5.3 Data processing](#_nvvpnfg8uzqm) 12

[3.5.4 Data analysis methods](#_a5sfci7s05he) 13

[**4. Results**](#_s00n078n8qan) **13**

[4.1 Supervised Classification with Machine Learning](#_49z7sxtvn7rk) 13

[4.1.1 Hypotheses](#_8atz01c3pcq1) 15

[4.1.2 Grid Search and Cross Validation](#_xnjzb22uuqxo) 15

[4.1.3 Machine Learning Outcomes](#_6n291i51zp2) 16

[4.1.4 Global Applicability?](#_rjdqzwfau99m) 16

[4.2 Thresholding](#_r0p6ydtb4fcu) 17

[4.3 Image differencing of SAR imagery](#_osj6j6g5tvm) 17

[4.4 Urban flood detection with time series analysis](#_aohugcui91al) 18

[**5. Discussion**](#_eldxlv5nszax) **21**

[**6. Conclusions and Recommendations**](#_trhg7ka9rxc6) **23**

[6.1 Conclusions](#_pobqqwpvvb90) 23

[6.2 Recommendations](#_maddjswdr84v) 23

[**References**](#_njnj4tl5w96y) **25**

[**Appendices**](#_nq7lz8mga6us) **27**

[Appendix A - Workflows](#_re81b4rsmk91) 27

[Appendix B - Machine Learning: Grid Search Parameter Settings](#_6o56ipkur5yo) 28

[Appendix C - Machine Learning Results](#_7eal3npds908) 29

[Cap-Haitien](#_3a2fvfqvrnp2) 30

[N’djamena](#_1tssss2507vg) 38

[Appendix D - Overview of files](#_boc5lykts0gl) 46

[Appendix E - Additional result of image differencing](#_qc5cf7k7ehff) 48

# 

# Executive summary

In recent years, climate change has become an important factor for conflict. To mitigate these conflicts and to find suitable locations to set up camps, information about natural disasters, such as floods, is valuable to the Ministry of Defence. Therefore, in this project a time series analysis method will be created showing the spatio-temporal characteristics of flood events. This report serves to give the reader an intensive background on the methods used in the project and the choices that were made.

In this report multiple methods of time series are explored based on the use of sentinel-1 SAR data. These methods are thresholding, random forest, k-nearest neighbour, gaussian naive bayes classifier, support vector machines, average backscatter and difference maps. The last two methods were specifically created to detect urban floods, because urban floods could not be detected easily by the methods that were created. This resulted in two parts of the model: one for urban floods and one for non-urban floods.

The machine learning methods turned out to be the best methods to classify floods. From the machine learning methods, k-nearest neighbours is the most accurate, but all of these methods do not differ that much in accuracy. Both machine learning and thresholding methods had difficulties detecting floods in urban areas, whereby two other methods were created: image differencing of SAR imagery and a time series analysis. The time series analysis had the best results and the image differencing method was found to be too inaccurate, therefore its use is not recommended. In conclusion, machine learning methods are recommended to use. However, due to the different backscatter in different countries and areas, polygons need to be created by the user to use as training data. When this is done, the machine learning method could be applied globally.

# 

# 1. Introduction

## 1.1 Background

The Dutch Ministry of Defence is responsible for the protection of freedom and peace, both on a national and international level. Besides, the Ministry provides humanitarian aid and maintains (inter)national rule of law whenever natural disasters or accidents occur. The effects of climate change are a growing threat to the freedom and peace of many nations, and its impact is becoming more frequent and severe. For example, drought can lead to conflicts over access to water, while the sea level rise and flood events can displace communities or even whole nations. Hence, ongoing climate change is accompanied by a growing need for military support to manage and solve such conflicts and protect communities worldwide.

Knowledge about the environment and the magnitude of disasters is vital, not only before but also during military operations (Defensie, 2021). Geo-information systems (GIS) and remote sensing techniques can give insight into the environment and terrain of the area of interest, without the need to collect data on-site. Time series analysis can be used to gain knowledge about changing spatiotemporal patterns and to serve as input for the preparation phase of humanitarian operations.

The objective of this project is to gain early insight into spatial temporal patterns of flood prone areas by creating an uniform and globally functioning time series analysis method. The primary goal of this method is to infer on a detailed scale the flood risk of certain areas where the Ministry of Defence is active.

## 1.2 Problem analysis

The relation between climate change and conflict has been studied extensively during the past decade. While climate change may not be a direct cause of conflict, there is strong evidence that it indirectly increases the risk of conflict and violence (Koubi, 2019).

Extreme weather conditions, such as storms, drought, heat waves and floods can worsen existing social, economic and environmental factors (Nordås & Gleditsch, 2015). These extreme weather conditions can also affect the water and food security of entire regions by for instance decreasing agricultural productivity. They can cause famine, disease outbreaks and lead to competition for scarce resources. Ultimately, this can result in population displacement and migration flows, possibly also impacting surrounding countries. Migrants could encroach national or foreign territory where the living conditions might also be poor (Nordås & Gleditsch, 2015). Moreover, conflict and migration can form a breeding ground for terrorism as certain groups could take advantage of the vulnerable position of displaced people in need for help (Defensie, 2022).

This project can assist the Ministry of Defence by exploring possibilities for the risk assessment of floods. For instance, knowing the extent and the frequency of floods can help military and rescue operations by efficiently setting up forward operating bases.

## 1.3 Objective

The main objective of this project is to gain early insight into spatial and temporal patterns of flood prone areas by creating a time series analysis method. This concerns for instance:

* Flood patterns;
* Magnitude;
* Duration;
* Flooding recurrence time.

Flood risk assessment is necessary to develop appropriate measures to locate forward operating bases on safe grounds, mitigate flood risk, evacuate people in time and protect infrastructure. Several remote sensing techniques have been studied within the context of disaster management, for instance using synthetic aperture radar (SAR) (Matgen et al., 2011) and infrared imaging radiometry (Lacava et al., 2019). This project explores the possibilities of applying remote sensing techniques and time series analysis to SAR data for flood risk assessment.

Another objective is a workflow model of our conducted methods so the commissioner can understand and determine how they can incorporate the recommendations of each method. It is therefore vital to communicate the recommendations, as well as the constraints and restrictions we have found while conducting the different methods.

The developed workflow should be globally applicable. Hence, globally available and most preferably open-source data should be used. Moreover, the amount of data used as input should be minimised since importing and scrubbing data is a time intensive task due to strict security guidelines at the Ministry of Defence.

The objectives of this study are:

1. Exploring the possibilities and processes of analysing to floods with the use of remote sensing;
2. Determining restrictions, (pre-conditions) and constraints of the analysis;
3. Creating the time series analysis model.

To create and test each individual method, the city of Cap-Haïtien in Haïti has been chosen as the main study area, primarily since the majority of the floods were evident from the SAR data. However, the model is built to be globally applicable so other study areas can also be used in future research.

# 2. Materials

Before explaining the conducted methods in this research in more detail, the different datasets are discussed.

## 2.1 Synthetic Aperture Radar (SAR)

The first dataset consists of radar data. The use of synthetic aperture radar (SAR) for disaster management has been studied extensively as it provides operational advantages when compared to optical and infrared sensors. SAR can operate independently of sunlight and of almost all weather conditions, which means it secures the acquisition of useful data on a consistent basis. SAR is therefore particularly useful when monitoring flood events, as the areas where floods occur are often characterised by persistent cloud cover (Matgen et al., 2011).

The accuracy of deriving flood areas from SAR imagery depends on several factors. Firstly, it depends on the signal parameters (such as polarisation, wavelength and incidence angle). Polarisation refers to the plane in which the transmitted electromagnetic radiation oscillates.

Long et al. (2014) posit that HH (or horizontally transmit, horizontally receive) polarisation is the preferred polarisation for flood mapping, as it has a lower sensitivity to waves on the water surface of flooded areas. However, Ezzine et al. (2018) found VH polarisation to be superior for flood mapping. Using HH (and VV) polarisation would, however, also imply that deriving flooded areas within urban settlements can be complicated due to the double bounce scattering (mainly characterised by HH) off buildings (Matgen et al., 2011). Furthermore, buildings and taller vegetation could cause shadow or layover that can make water appear invisible on radar images (Mason et al., 2009).

The SAR data in this project has been acquired by Sentinel-1, as this is the only high resolution satellite of which data is still acquired and made openly available. The data solely concerns VV and VH, as Sentinel-1 mainly operates in this dual-polarisation mode. Data in HH and HV is only acquired over the arctic and very few other selected sites. The VV and VH data is downloaded as ‘’decibel gamma0, radiometric terrain corrected.’’ Gamma0 (or gamma nought) refers to one of four radiometric calibration outcomes (together with the digital number, beta0 and sigma0). Gamma0 is different from the other calibrations as it takes into account the local incidence angle, or topography. Radiometric terrain correction refers to the two main types of correction that are performed using a digital elevation model and that aim to enable intercomparison between SAR images acquired on different dates or in different modes (Small, 2011). Firstly, terrain correction concerns projecting the pixels from the satellite reference system onto a geographic reference system and correcting for geometric distortions caused by the geometry of the radar antenna and the topography. Secondly, radiometric correction concerns correcting for the intensity of the backscatter of pixels which are distorted by the local incidence angle. The ‘’gamma0 radiometric terrain corrected’’ product involves a terrain flattening algorithm that minimises the effect of the terrain (Small, 2011).

## 2.2 Digital Elevation Model (DEM)

Because height has a big influence on which areas get flooded in a certain region, this is also an important factor to take into account when predicting flood patterns. Hence, a Digital Elevation Model (DEM) is used to aid in the supervised classification with machine learning.

## 2.3 Water bodies

The third dataset is permanent water bodies, which is mainly used for the supervised classification with machine learning (see Chapter 3). The water data that should be masked out is derived from the website “Global Surface Water”. The dataset is developed by the European Commission’s Joint Research Centre in the framework of the Copernicus Programme (European Commission’s Joint Research Centre, 2022). Location and temporal distribution of water surfaces of the world are stored, just as statistics on the extent of the water bodies and the seasonal shifts. The dataset is based on Landsat imagery.

## 2.4 Global Human Settlement

The Global Human Settlement (GHS) dataset contains several data layers that show the human presence on the planet over time, specifically focussing on settlement and population density (European Commission, 2019), which is another valuable variable for flood prediction. Not only because of the consequences on the people living in flooded areas, but also since floods show different backscatter patterns in urban and rural areas. By adding the GHS as a prediction layer, models could achieve better performance in making a distinction between flooded and non-flooded urban areas. More information about the specific layers are found in later sections in this report.

## 2.5 Precipitation

Another source of input is precipitation data from the NASA Power project (NASA POWER, 2021). NASA POWER provides high resolution precipitation data that is derived from NASA’s Global Precipitation Measurement (GPM) mission’s Integrated Multi-satellitE Retrievals for GPM. The precipitation data has a global resolution of 0.1° x 0.1° latitude/longitude grid (approximately 10km). GPM uses multiple ground and space measurement instruments to improve the quality of the precipitation database (Global Precipitation Measurement, 2022). This dataset is used for the last method where urban backscatter values are compared with local precipitation data.

# 3. Methods

The main objective of this report is to investigate the potential of combining remote sensing techniques with time series analysis for the risk assessment of floods. To investigate the potential, several methods are conducted. The methods can be classified into four main groups, which are:

1. Supervised classification with machine learning
2. Histogram thresholding
3. Difference maps
4. Urban time series analysis of SAR and precipitation data

This chapter is divided into 5 subchapters. The first chapter gives an overview of the different datasets that are used for the four methods. Besides, a brief description of downloading the SAR data from either the EO Browser website or the EO Learn API script is given. Chapter two to five explain the workflow of each of the four methods, following the order as defined above. The workflow of each of the four methods, following the order as defined above, are described at subchapter 4.2 to 4.5. In these chapters, the pre-processing and loading, the preprocessing, and the data analysis steps are described more thoroughly.

## 3.1 Data acquisition

Before explaining the pre-processing steps and each method more thoroughly, several datasets should be downloaded. Data acquisition can be done either directly from the internet or indirectly by running the Application Programming Interface (API) scripts. A prerequisite of each proposed method is SAR data. Furthermore, method-specific datasets are also required to conduct the different methods. The datasets necessary for each method are summarised below:

1. **Supervised classification with machine learning models**
   1. SAR data
   2. Global Human Settlement Layer (GHSL)
   3. Permanent water bodies data
   4. Digital Elevation Model (DEM)
2. **Histogram thresholding**
   1. SAR data
3. **Time difference maps**
   1. SAR data
4. **Urban time series analysis of SAR and precipitation data**
   1. SAR data
   2. Global Human Settlement Layer (GHSL)
   3. Temporal precipitation data

Two API scripts are created to download the datasets from the internet; the first for downloading SAR data from Sentinel Hub, and the second for downloading precipitation data of the area of interest. Data acquisition with the former API is explained in chapter 4.1.2, and the latter API is described in more detail at the urban time series analysis method.

### 3.1.1 Sentinel Hub EO Browser

The first method to download the SAR data is from the Sentinel Hub EO Browser website. This website allows visualising and downloading satellite data from numerous satellites and data archives. When downloading the data from the EO Browser website, only the “decibel gamma0, radiometric terrain corrected” VV and VH data should be downloaded (for more info, see 3.Methods). Besides, the SAR data from EO Browser should be processed and stacked, which is done with the “LoadAndStackSentinelData.py” function. A detailed description of downloading the SAR data from EO Browser is given in the “InstructionsForPreperations.pdf” file in the main folder.

### 3.1.2 EO Learn API

The API to download SAR data, saved in the folder ‘eo\_learn’, is an alternative method to download the SAR data from Sentinel Hub. When using the EO Browser website, the SAR images have to be downloaded individually for each date. Since temporal data is a prerequisite for time-series analysis, it is a time-consuming task to use the EO browser website to download SAR data for a longer time period. Hence, the EO Learn API script has been created which automates the process of downloading temporal SAR data. The input parameters of this script are the start and end date, the coordinates of a bounding box around the area of interest, the resolution, and the output folder. All data obtained from the API, consisting of VV, VH, VV/VH ratio and the DEM, are stacked and processed according to the CARD4L processing standards, and finally saved to the data folder. Hence, when downloading the data from this API, it does not have to be unzipped, processed, and stacked with the “LoadAndStackSentinelData.py” function in the main folder.. A more detailed description is given inside the script and the “InstructionsForPreperations.pdf” file.

## 3.2 Supervised Classification with Machine Learning

A graph explaining in detail the workflow of the machine learning algorithms can be found in appendix A.

### 3.2.1 Training Data and Processing Steps

Four supervised machine learning methods were selected for this project. Training data was created to train the models. A more elaborated explanation about how to create training data can be found in the file “*Instructions for Running the Time Series Application*”.

The SAR data is already radiometrically corrected and orthorectified when it is downloaded. The only preprocessing of the SAR data consists of a speckle filter to remove unwanted variance in the image. The GHS data and the DEM are cropped to the extent of the SAR image and resized to exactly match the size of the SAR data. In order to improve the distinction between flooded land and dry areas, an index was calculated based on the VV and VH backscatter:

All inputs are rescaled to a range from 0 to 1, since this is necessary for machine learning classifiers that use the distance between pixels in the feature space for training.

### 3.2.2 Grid Search and Cross Validation

The Scikit Learn package provides a large amount of functionalities for machine learning. The script uses grid search and cross validation functions to find the optimal parameter settings based on model performance on validation data. The grid search function applies a custom loss function to assess the model performance. This loss function is based on the F-beta score, which is a harmonic mean of precision and recall, of the two classes of interest: flooded land and flooded urban. For each candidate, 5-fold cross validation is used to produce more robust estimates of the loss associated with each candidate. The script saves the grid search outputs in a .csv file for future reference.

### 3.2.3 Gaussian Naive Bayes Classifier (GNB)

The GNB algorithm is based on Bayes’ theorem, which describes the probability of an event, given prior conditions. For each test sample, it calculates the probabilities for each class and assigns the class with the highest probability. The classifier is known for the fact that it is a lot faster than other popular classifiers, especially compared to support vector machines and random forest (Ontivero-Ortega, et al., 2017). Apart from its relatively short computation time, the algorithm is also easy to implement, since it is non-parametric and therefore does not require any tuning of hyperparameters.

### 3.2.4 K-nearest neighbours (kNN)

Another machine learning method is kNN which is also known as a simple and fundamental method for machine learning classification (Peterson, 2009). The algorithm assigns to a test sample the class that the majority of the N closest neighbouring training samples belong to, where closeness is defined by the Euclidean distance between a test sample and training samples. The simplicity of the model is mainly defined by the fact that it only has one hyperparameter.

### 3.2.5 Support vector machines (SVM)

A SVM-classifier constructs decision boundaries (separating hyperplanes) between classes, splitting the feature space into different regions per class (Noble, 2006). However, classes are often not perfectly separable. This is why the SVM-classifier allows for a number of samples to be classified wrong, as specified by the user. The SVM-classifier can use different types of nonlinear kernels in order to solve complex, nonlinear classification problems. A SVM-classifier with a linear kernel is the most simple and is equivalent to a support vector classifier (SVC). A SVM-classifier with a polynomial kernel is more powerful in solving nonlinear problems, depending on the degree of the kernel as specified by the user. Alternatively, the radial basis function kernel can be used, which is the default option. In short, SVM-classifiers are a set of classifiers that can solve complex classification problems, but it also requires some more tuning of hyperparameters.

### 3.2.6 Random Forest (RF)

The last machine learning algorithm used for classification in this project is RF. RF is an ensemble method that builds *N* decision trees, each of which is trained with a random subset of the training data and validated with the remaining (out-of-bag) samples. Each test sample goes through each decision tree and is therefore classified *N* times. Ultimately, the majority rule is used to assign one class to each test sample. RF is a popular algorithm that usually performs well without much configuration. However, there it has a lot of options that the user can tweak, such as the maximum depth, the minimum number of samples at each leaf node, and more.

### 3.2.7 Predictions and Flood Frequency Analysis

After the classification of the radar images, permanent water bodies are masked out from each prediction image. A sieve filter is applied to remove isolated specks from the raster. This proved to be useful to reclassify pixels that are wrongly classified as flooded as dry areas. Then, all prediction maps are combined to create a flood frequency map. This shows how often each pixel is flooded. Hence, it provides an estimation of the flood risk of each part of the area.

## 3.3 Histogram thresholding

Thresholding, also known as histogram thresholding, is a relatively simple method for image classification. The method separates an image into multiple classes based on peaks in the histogram of the image (Deshmukh & Shinde, 2006). In this case the classes are defined as flooded and non-flooded. A threshold between the two or more classes is then defined by the valleys between the peaks (Long, et al., 2014). In this project, KMeans clustering is applied to find the centre of both clusters and the midpoint between both centres is considered to be the threshold. In an ideal case, the histogram of backscatter intensity follows a bimodal distribution where one peak indicates water and the other indicates land. When the SAR image barely contains permanent water bodies the distribution will, however, be different and the clustering will already work less precisely. Another major limitation of this method is that it may not map urban areas that have been flooded, as the VV backscatter in such areas can increase instead of decrease.

After determining the threshold all values below it are classified as flooded, and all values above it are classified as non-flooded. This obviously can be considered a gross oversimplification. Still, it is deemed interesting to involve this simple method to show the difference with more sophisticated methods.

## 3.4 Image differencing of SAR imagery

The creation of different maps is a rather easy way to detect floods in urban areas. Because classification of urban floods was found out to be quite difficult, this was thought of as a solution. With image differencing of SAR imagery, difference maps are created by subtracting consecutive maps from each other or from one specific map, which results in a map showing the differences between the two days. Because only the urban areas were wanted and only the high increase in backscatter can be assigned to urban floods, the use of a threshold was implemented. The script is written so that it subtracts every image from the first image, because otherwise it sees a high backscatter in areas which declined in backscatter before, but then restored again. The threshold is a fraction which will be multiplied with the max value to get a threshold which is applicable on a global scale. Next, all values above this threshold are assigned with the value 1 and everything below this value is assigned with the value 0. The last step is adding all these maps to create a flood frequency map of the urban areas.

## 3.5 Urban flood detection with time series analysis

This method explores flood detection with time series analysis of urban areas. This is done by calculating the mean VV backscatter for urban areas and comparing these backscatter values with precipitation data of the same area of interest. Moreover, to find anomalies inside the two datasets, both visual and statistical outlier detection methods are used.

The idea behind this method is to investigate whether there is a relationship between precipitation data and VV backscatter data in urban areas. Our hypothesis is that there is a positive relationship between precipitation and VV backscatter in urban areas, assuming that an increase in precipitation results in a higher likelihood of floods. If this applies, a flooded urban area will have higher VV backscatter values than non-flooded urban areas (due to the ‘double bounce’ effect). Hence, when using time-series analysis, it should be possible to detect a relation between the anomalies in the precipitation and SAR data. In other words, higher precipitation should be accompanied by higher VV backscatter values in urban regions.

The explanation of the workflow of this method is briefly described below. A graph that explains in more detail the workflow can be found in Appendix A. To run this method, three datasets are required (for more info, see “InstructionsForPreperations.pdf”):

* Stacked and preprocessed SAR data
* Precipitation data (csv file with daily precipitation)
* Urban area raster (downloaded from GHSL)

### 3.5.2 Data preprocessing and loading

The SAR data should be loaded and preprocessed whenever it is downloaded from the EO Browser website. When it is downloaded with the EO Learn API, the loading and preprocessing has already been done. The preprocessing and loading step for the daily precipitation data is also incorporated inside the script; the average precipitation for the bounding box is exported to a .csv file, as well as the precipitation sum of the last x days. Since floods often occur due to the accumulation of precipitation over several days, each date also contains a column where the precipitation of the last x days are summed. The number of days that should be summed should be defined inside the script, as well as the bounding box for which the precipitation data is downloaded. Moreover, the urban raster is converted to a GeoPandas Dataframe so it can be used to clip the SAR data for each date.

### 3.5.3 Data processing

The mean VV backscatter values are calculated for the clipped SAR images per date. These mean values are written to a list and exported to the ‘data’ folder as a .csv file, displaying the mean urban VV values for each date. Since both the precipitation and mean VV files contain a ‘date’ column, they can be merged into one large dataframe.

### 3.5.4 Data analysis methods

At this step in the workflow, several plots are made with the large dataframe that contains the precipitation and mean VV data. These are:

* Line plots where the mean urban VV backscatter values are plotted against the date, and the average precipitation and the sum of the precipitation for each given date
* Histograms showing the count of the different VV and precipitation values
* Boxplot showing the distribution of the data based on five criteria (minimum, first quartile (Q1), median, third quartile (Q3), and maximum. This is done for both columns (mean VV and precipitation).
* Scatterplot where the urban mean VV backscatter is plotted against the precipitation data.

These plots are used to visualise the relationship between the precipitation and the VV backscatter. From these plots, it is possible to visually detect outliers within the dataset.

Rather than visually investigating the

Besides, to statistically prove that there are outliers in the data, five statistical outlier detection methods are used. More background information and the specific parameters used for each method can be found inside the script itself (“urban\_areas”→“py”→ “s23OutlierDetection.py”). These outlier detection methods are:

1. Thresholding
2. Tukey's method (probable)
3. Tukey’s method (possible)
4. Z-score
5. Median Absolute Deviation (MAD)

After conducting the outlier detection methods, the line plots and scatterplots are returned where the outliers are highlighted inside the plots. Hence, the output data is a collection of plots that are created after the outlier detection methods.

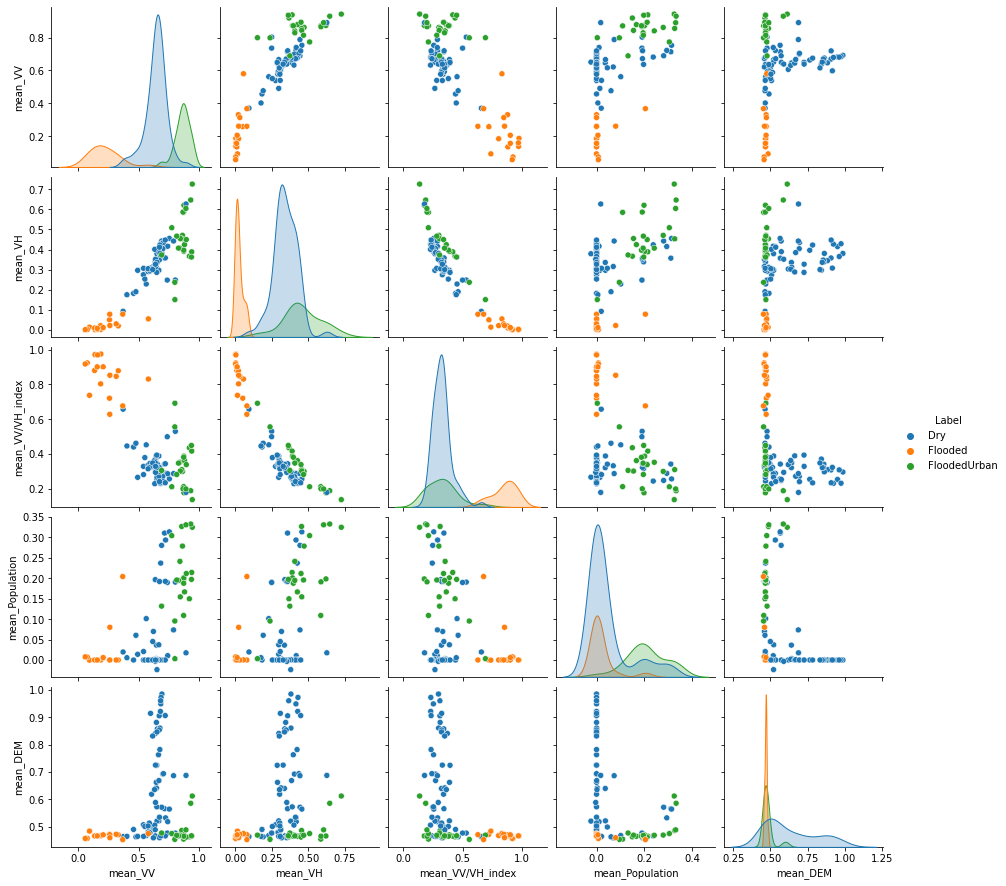
# 4. Results

## 4.1 Supervised Classification with Machine Learning

The machine learning script was tested on two study areas: Cap-Haitien in north Haiti and N’djamena in southwest Chad.

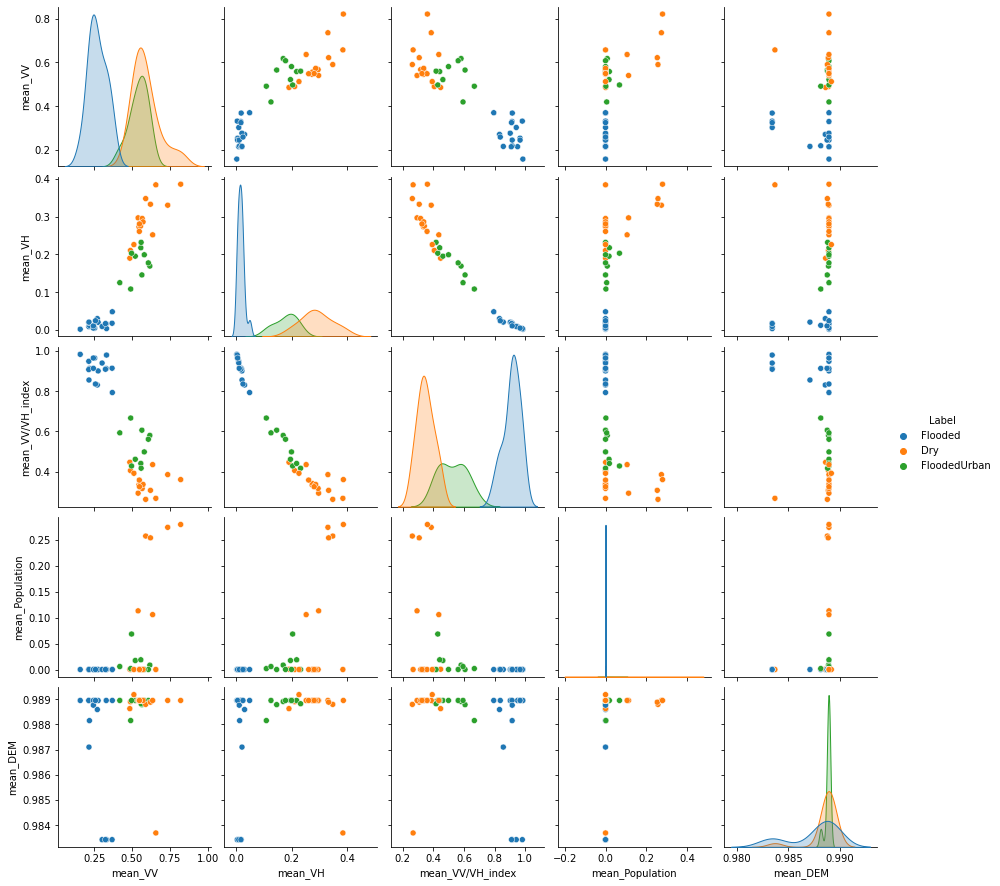
For Cap Haitien, a time series that consisted of 39 Sentinel-1 images between 01-01-2021 and 26-04-2021 was used. In this time, the area was subject to numerous flood events, most notably on 05-04-2021, when the city of Cap-Haitien was severely affected. Training data was created from four dates in this timeframe, two of which included a flood event. A total of 137 polygons on four separate dates were created on different spots in the area that were classified as dry, flooded land, or flooded urban areas.

Figure 1 shows that the different classes of the training data can be well distinguished. Flooded land in particular has lower backscatter values than the other two classes in both VV and VH polarisation, which also results in higher index values. Flood events only occur in low areas and urban floods occur solely in areas with high population values. The graph implies that dry pixels have the most variation and that the most difficult separation would be between the dry and flooded urban classes.



*Figure 1 - Pairplot showing the pairwise relationships in the dataset of Cap-Haitien*

For N’djamena, a time series of 17 Sentinel-1 images between 04-08-2020 and 08-11-2020 was used. Multiple flood events occured in this time frame, but ground truth data of affected areas is hard to find. Training data was created of a flood event on 02-11-2020 in the area, which affected both parts of the city and rural areas around the city. Training data was only created for one date and consisted of 68 polygons. Figure 2 shows that the different classes are more difficult to separate for N’djamena than for Cap-Haitien, as there appears to be more overlap between the classes.



*Figure 2 - Pairplot showing the pairwise relationships in the dataset of Cap-Haitien*

### 4.1.1 Hypotheses

For the study area of Cap-Haitien, considering the small number of predictors compared to the number of training samples and the relatively clear distinctions between classes in the feature space (Figure 1), it is expected that the simple algorithms (GNB, kNN) will perform as well on the data as the more complex algorithms (SVM, RF). Besides, it is expected that the grid search will favour less complex candidates that are less prone to overfitting than more complex candidates.

For the study area of N’djamena, more complex models are probably needed to be able to distinguish the different classes, since there appears to be more overlap (Figure 2). More complex models could be favourable, but could also increase the risk of overfitting.

### 4.1.2 Grid Search and Cross Validation

It was found that there is no use to train a common, globally applicable model with training data from different regions, since there is a large variation in backscatter values between different regions. For example, a flooded urban area in Cap-Haitien and a flooded urban area in N’djamena have different backscatter values due to differences in building density. Therefore, it was chosen to train models for each study area separately.

A grid search of different parameter settings is performed when calling the functions to train models that require some hyperparameters (RF, SVM, kNN). Appendix B gives an overview of the different parameter settings (candidates) that were tried and the optimal candidates that were found for both study areas.

The grid search results are different for each study area. For Cap-Haitien, the grid search favoured less complex candidates. For the SVM, a linear kernel with 2nd degree and a low value for the regularisation parameter were found to be optimal. For the RF, a minimum number of samples of 3 per leaf and a maximum depth of 10 were found to be optimal.

For N’djamena, more complex models were favoured, presumably due to the more complex nature of the data.

### 4.1.3 Machine Learning Outcomes

Appendix C includes for each study area and for each model a selection of metrics, one prediction image for a day without flooding, one prediction for a flood event and three frequency maps. The results were obtained with a global random seed, such that they are reproducible. It must be mentioned that different seeds will lead to different results, as all models have some random component in their training phase.

For Cap-Haitien, the results are generally positive, as the model performances on the test data and the classification results are all very similar. A high test accuracy of 0.91 is obtained by the GNB, SVM and RF classifiers, while the kNN classifier reaches a test accuracy of 0.94. All models managed to get perfect scores for the user’s and producer’s accuracy of flooded land, which shows that this class is indeed easy to separate from the other classes. However, a visual inspection of the prediction results and the frequency maps shows that the airstrip of the airport is wrongly classified as flooded land in all predictions. The training polygons that were located on the location of the airstrip were probably part of the training dataset. All models had more difficulties with the classification of flooded urban areas. The kNN model is also the superior model with respect to the user’s and producer’s accuracy of this class. However, it still wrongly classifies the most densely built-up parts of the city of Cap-Haitien as flooded in all images, just like the other models.

Overall, it can be concluded that the machine learning classifiers that were tested in this project all perform well on the data that was extracted from the training polygons. There are some general pitfalls, which could be solved by creating more and more accurate training data.

For N’djamena, the results are less positive. The kNN-classifier was again found to be superior, but the overall accuracy amounts to 0.84. User’s and producer’s accuracy of flooded urban areas are much lower at 0.67 and 0.4 resp. Hence, it is no surprise that the frequency and prediction maps imply that the estimated number of pixels classified as flooded urban is greatly overestimated. One reason could be severe overfitting, due to the lower amount of training data.

### 4.1.4 Global Applicability?

One goal of this project was to research the potential for a globally applicable time series method. The results of the supervised classification with machine learning imply that this is not entirely possible, due to large differences in backscatter values of each class. This is caused by differences in, amongst others, vegetation, terrain type, land use and building types. This problem was solved by only using a model that is trained and validated with training data from a specific region to classify images of that region.

The results also imply that different regions and different data lead to different optimal parameter settings. In this project, kNN with a similar number of neighbours was superior in both study areas. However, it could be that in future research, it appears that a different model performs better.

## 4.2 Thresholding

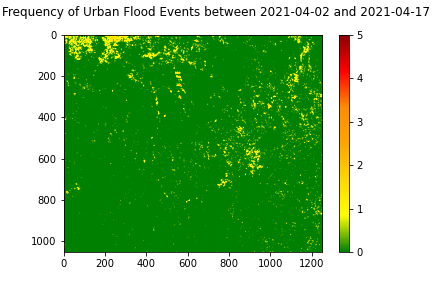
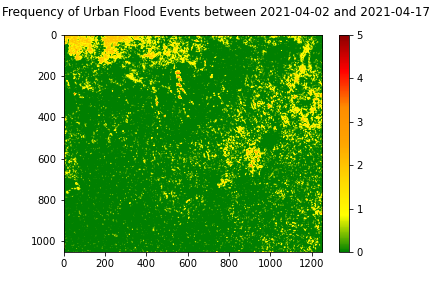


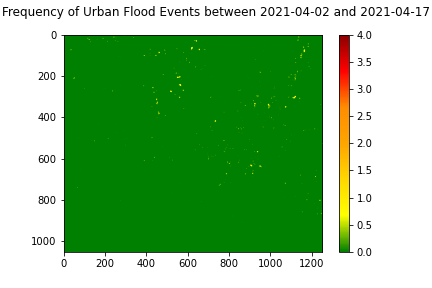
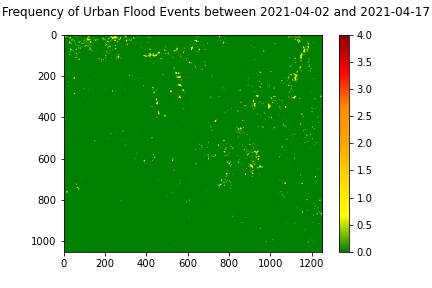
*Figure 2 - Classification outcomes of the thresholding method*

Thresholding is a fast and simple method to classify flooded land, but not flooded urban areas. This can be seen in Figure 2. The flood is clearly visible on the fourth of february (the flood was reported to have happened one the first). The city of Cap-Haitien, however, does not appear flooded while it actually was. The fact that the image of March 31st shows more ‘flooded’ land than the image of april 7 also indicates the inferiority of this method. It is assumed that the effect of the different orbits of the satellite (which are greatly minimised by radiometric correction) is the reason for this unusual pattern.

## 4.3 Image differencing of SAR imagery

For the results of the image differencing of SAR imagery, the city Cap-Haitien is chosen as the study area, which lies on the top right corner of the flood frequency map. The team chose to select a larger area, not only the city, to be able to see if other areas are marked as urban floods as well. The model has been run twice: first with 6 images between the 2nd and 17th of april (2021), and second with 100 images between the start and end of 2020. The threshold is changed to show how important and variable this parameter is. Only the “VV” polarisation band is considered since this band is responsible for the double-bounce effect in flooded urban areas. The results are four flood frequency maps.





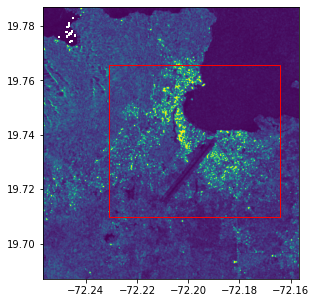
*Figure 3 - Flood frequency maps of urban areas which is created by image differencing of SAR imagery of 2021-04-02 until 2021-04-17. Top left: threshold = 0.3, top right: threshold = 0.4, bottom left: threshold = 0.5, and bottom right: threshold = 0.6*

In Figure 3 it is visualised that the threshold of 0.5 is most suitable for image differencing of SAR imagery. This is because with a threshold of 0.5 a lot of areas are not wrongly marked as flooded and there are still urban floods that are marked as urban floods. During this period of six images, there was a flood, but the model does not completely include the extent of the flood or includes other pixels as well. This method is therefore quite inaccurate.

## 4.4 Urban flood detection with time series analysis

The results of the urban flood detection with VV backscatter and precipitation data are explained in more detail in this subchapter. As mentioned in Chapter 4.5, a time series analysis is used to find a potential relationship between VV backscatter and precipitation data in urban areas. Both visual and statistical outlier detection methods are used to find anomalies in the dataset which indicate floods of urban areas.

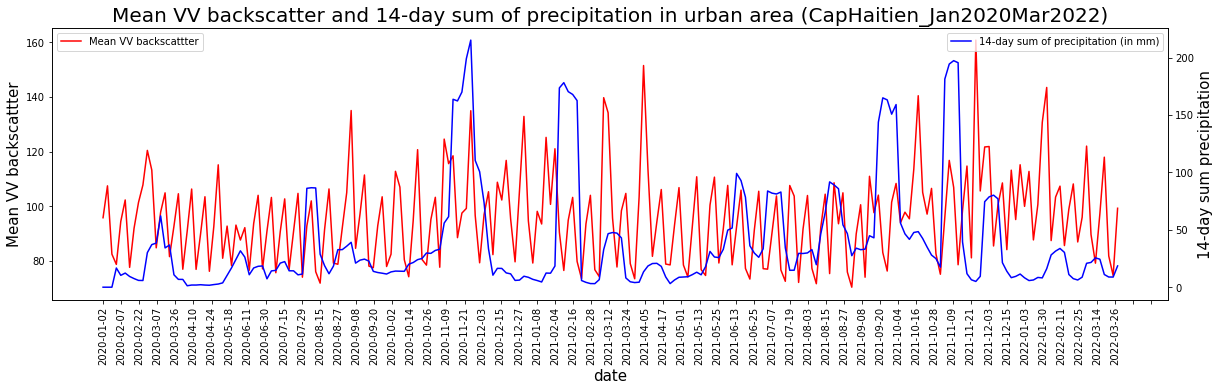
To display the results, SAR images and precipitation data between January 2020 and March 2020 of Cap-Haïtien (Haïti) are taken. The bounding box that is used to clip the urban raster data with the SAR data is shown in Figure 4 (left), as well as the first SAR image that has been clipped by the urban extent (right).

**

*Figure 4: SAR image of Cap-Haïtien (Haïti) with a bounding box where the urban areas will be clipped from (left), and clipped SAR image by urban areas (right).*

Then, the mean VV backscatter of the clipped urban areas is calculated for each individual date within the dataset. The mean VV backscatter values per date are plotted in Figure 5 (red) as a line plot, as well as the 14-day precipitation sum for each date (blue). Before comparing both lines, it is worth mentioning that there is a distinctive zigzag pattern in the mean VV backscatter graph. Relatively higher values are immediately followed by lower values. This pattern can largely be assigned to the orbit direction of the Sentinel-1 satellite. The dataset consists of a mixture of SAR images with an ascending and descending orbit direction, where the ascending images have on average a higher VV backscatter than the descending images.

As one can observe, both the mean VV backscatter and the precipitation have a few distinctive outliers. A few of the peaks in the 14-day precipitation sum are closely followed by peaks in the mean VV backscatter, for instance at the 9th and 25th of November in 2021. However, this relation is often not clearly evident when comparing the precipitation and VV backscatter lines.



*Figure 5: Line plot displaying the mean VV backscatter for urban areas in Cap-Haïtien (red) and the 14-day precipitation sum (blue) plotted against the date*

To statistically determine whether the dataset contains outliers, and if so, which dates are considered as outliers, several statistical outlier detection methods have been conducted (see Chapter 4.5). The scatter plots after the outlier detection are shown in Figure 6, where the outliers for both the precipitation (left) and mean VV backscatter (right) are highlighted according to the outlier Likelihood Ratio (LR). The LR states how likely it is that a data point is an outlier according to the 5 detection methods (i.e., LR of 0.4 means that 2 of the 5 methods have classed this point as an outlier).

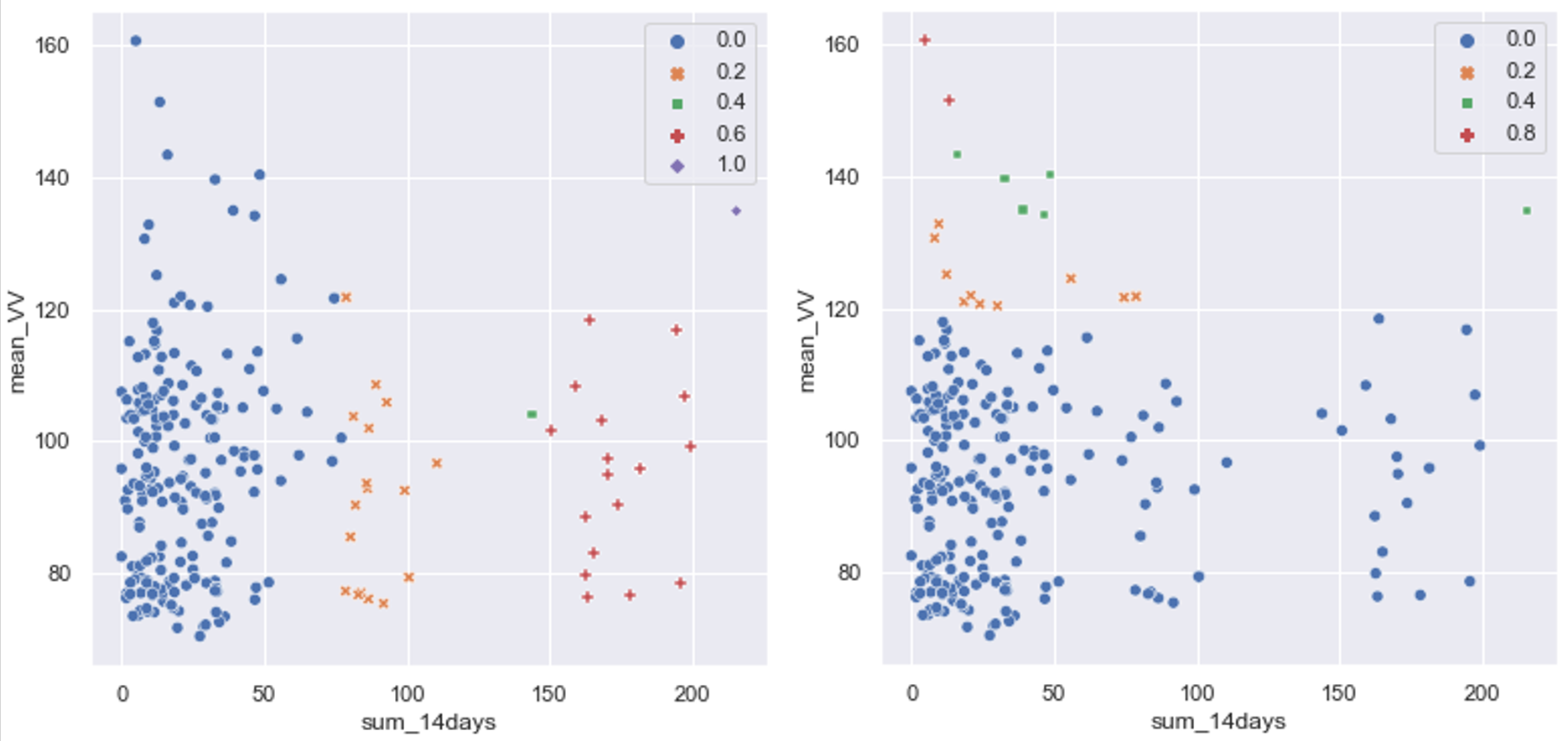
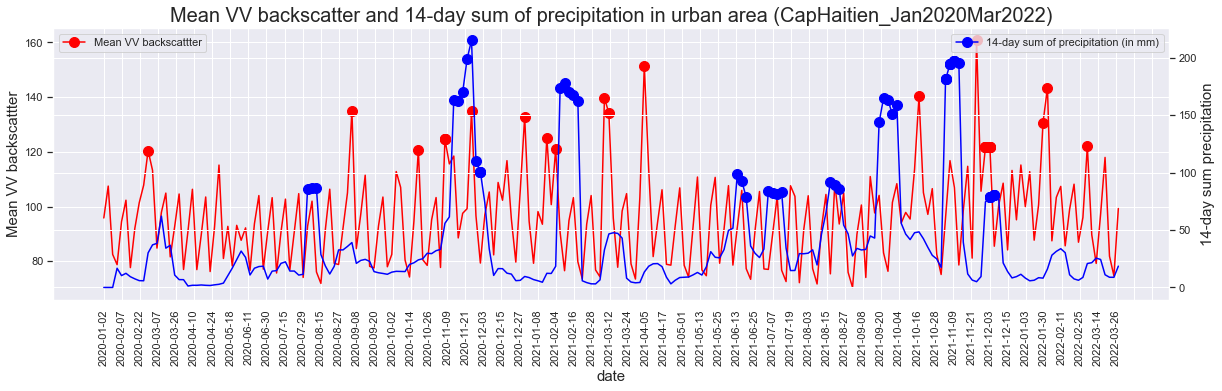
*Figure 6: Scatter plots where mean VV backscatter is plotted against the 14-day precipitation sum. Outliers in the precipitation data (left) and mean VV backscatter data (right) are highlighted based on outlier Likelihood Ratio.*

Figure 7 shows the precipitation and backscatter outliers on top of the line plots of Figure 5. The outliers of the precipitation data are much more clustered than the anomalies of the urban VV backscatter outliers. Besides, not every outlier of the urban VV backscatter data is preceded by precipitation outliers, and only a few precipitation and backscatter outliers are within a timeframe of approximately 20 days.



*Figure 7: Line plot displaying the mean VV backscatter for urban areas (red) and 14-day precipitation sum (blue) plotted against the date. Outliers are highlighted with their corresponding colour.*

# 5. Discussion

There are several problems we ran into during the project and the development of our application, which are explained in more detail in this chapter.

**Constraints from commissioner**

The Dutch Ministry of Defence cannot directly use the internet due to safety reasons. Besides, the ministry can only use specific programs because the ministry cannot confirm the safety while using it. Therefore, the project needed to be independent from the internet and created in programs approved by the commissioner. Because of this, all the data had to be downloaded manually, even though there are packages like the sentinel package in Python which could make it much easier. Moreover, some packages are only available in scripting programs such as R and not in python, whereby these packages cannot be used. For example, the BFAST package would be a good way to detect floods on maps, but cannot be used because it is only available in R.

**Data availability**

The model is based on the assumption that good open source data is available. The model runs on sentinel-1 data, which is openly available through EO Browser (EO Browser, 2022). When this website, in addition to other websites which provides open source data, does not work, the model cannot execute the time series analysis. However, the model should still work on other radar data, provided that the same polarisation and other settings are the same. Another data constraint is the global availability of data. In some areas, Sentinel-1 captures SAR images only once every 4 or 5 days. It is therefore possible that not every flood is captured by the satellite and that valuable SAR data is missing. Hence, the recurrence time of the SAR data needs to be taken into account when the results of the model are evaluated.

Moreover, there is a noticable difference between SAR images that have an ascending or descending orbit direction. As explained in Chapter 5.4, the VV backscatter values for urban areas are on average higher when the satellite is in an ascending orbit than when it is descending. It is therefore easier to detect a flood in ascending SAR images than descending images since the latter has generally lower values. This should also be taken into account, for instance looking at only ascending or only descending images. However, this will double the incurrence time of the available SAR images.

**Training data**

For training data, training polygons with three different classes are created, which are (1) Dry area, (2) Flooded area, and (3) FloodedUrban area. It was often difficult to create training polygons since not every flood is clearly shown in the data, and neither can the extent of the floods every time be found on the internet. For example, flooding in Haiti was clearly visible in 2021 but other flood events in this area were not always very distinctive. Hence, creating accurate training data can be a tedious task.

**Difficulties classifying flooded urban areas and forests**

The first methods that were used to classify an area as flooded or non-flooded were machine learning and thresholding. However, it became evident that classifying urban areas into different classes is difficult. Especially with thresholding, classifying urban areas was almost impossible. The reason for this is the difference in backscatter between urban areas and more or less smooth surfaces. Due to these difficulties, the team thought of other methods such as difference maps. Because these difficulties were used later in the process, there was limited time for working out these methods.

The same problem occurs with flooding in forests. Here the backscatter values also increase instead of decrease. However, unlike with cities it is not reported when forests flood. Therefore, it is very difficult to create training data and train models for forest flood detection. Because of this and because of time constraints it was therefore decided to just focus on land and urban areas and not on forest flooding.

**Image differencing of SAR imagery**

This method has been developed to investigate the potential of detecting floods in urban areas. The result shows some pixels which are classified as flooded, but it did not turn out to be highly valuable. It does not show the extent of the flood, but only some groups of pixels. This could be due to direct backscatter, so no double bounce, in larger parts of the city, whereby the effect of double bounce is declined. Moreover, there are some pixels that are wrongly assigned as an urban flood as well (Figure 3). In Figure 3, when the threshold is set to 0.6, there are a lot less pixels classified as flooded and floods did occur much less often. A good threshold is therefore very important. Where in Figure 3 there was most certain a flood and pixels were classified as flooded. But, in 2020 there are no floods found on the internet, while the results of image differencingstates that there were 10 or 20, depending on the threshold (Appendix E).

Therefore, it can be stated that the method of image differencing of SAR imagery is not suitable for a time series analysis of urban floods as an increase in backscatter can be caused by many more things than only a flood. The method could be improved which might lead to a suitable method for urban flood detection, but it is really hard to distinguish higher backscatter due to floods from other factors. Hence, other factors should be taken into account to improve this method.

**Effect of double bounce in urban areas**

The double bounce effect (i.e. high VV and HH backscatter) occurs whenever a SAR pulse hits a vertically-oriented object and is scattered back in the same polarisation direction (see Chapter 2.1 for more info). This effect is stronger in areas with many tall buildings. However, urban areas in third-world countries are often significantly different from urban areas in the western world, therefore influencing the significance of VV double bounce in third-world countries.

# 6. Conclusions and Recommendations

## 6.1 Conclusions

Several methods were explored and conducted within the scope of this exploratory research. Not every method achieved the same level of accuracy and usefulness. Supervised classification with machine learning turned out to be more accurate than the unsupervised thresholding method. However, it is difficult to adequately assess the quality of the supervised classification models, since the training and test data was not based on ground truth data, but created by the project members themselves and based on visual comparison of images from different satellite platforms.

The GNB, kNN, SVM and RF algorithms were implemented and tested, together with an automated grid search based on 5-fold cross-validation. The kNN-classifier proved to be the most accurate for both study areas. However, the differences in results among the different machine learning algorithms was small in the case of Cap-Haitien.

Both the machine learning and thresholding methods, however, posed issues regarding the detection of floods in urban areas. Hence, two separate methods were created with a focus on urban areas: image differencing of SAR imagery and a time series analysis of urban VV backscatter and precipitation data. From these two methods, the time series analysis returns the best result. From the several visual and statistical outlier detection methods, it can be concluded that there is some relation between urban VV backscatter and precipitation data. When visually inspecting the outliers, there appears to be some correlation between the two variables; above average precipitation data is often followed by above average VV urban backscatter. However, since validation data about the exact date and extent of the floods is missing, this correlation cannot be quantified.

One of the objectives was to create a globally applicable time series analysis. This has proven to be a rather complicated task. The patterns in backscatter intensities can greatly differ depending on the terrain, as well as the orbit direction of the satellite. This means a flood will also be characterised by different backscatter patterns depending on e.g. the country and the land use. The machine learning algorithms will therefore need training polygons of the area on which it is applied.

## 6.2 Recommendations

The aim of this project is to investigate the potential of gaining early insight into the spatial and temporal extent of flood-prone areas. After conducting several methods, we have made a list with recommendations for future research.

**Include forests in training data for machine learning method**

As mentioned briefly in the discussion both floodings in urban areas as in forests are difficult to detect. The urban areas were given their own methods to solve this issue, but the same could be done for forests to get a more complete overview of floods in a country. This can improve the result of the machine learning method since flooded forests, like flooded urban areas, return a high VV backscatter.

**Urban flood detection with time series analysis of VV backscatter and precipitation**

Several recommendations are defined for the urban flood detection method where the relation between the VV backscatter and precipitation data has been investigated. The first recommendation is to find ground truth data to validate our results. As of now, it is unknown what the exact start and end dates are of the floods in our study area, as well as the extent of the floods. Since the accuracy of this method cannot be quantified (i.e., validation data is missing), it is difficult to say what the exact relation is between VV backscatter and precipitation data in urban areas. Finding ground truth or validation data can therefore be used to quantify this relationship.

Another recommendation is to use precipitation data from the whole catchment area rather than a few coordinates around the area of interest. We expect that using this data will result in a more apparent relation between VV backscatter and precipitation. A flood can still occur in urban areas, regardless of whether there has been rainfall in the urban areas. It is much more interesting to look at all the rainfall that falls in the catchment area since floods often occur due to the accumulation of precipitation, not solely the precipitation that occurs in urban areas itself. Moreover, a rainfall-runoff model can be used to simulate how floods occur in a catchment area. However, runoff models are often not globally applicable but can nonetheless be useful to gain insight into the extent of floods.

The third recommendation for this method is to find a dataset that displays the urban areas more accurately, especially for third-world countries. When observing the urban raster datasets, several suburbs are not taken into account which has an impact when calculating the mean VV backscatter values for urban areas.

If time would allow, we would have stacked the clipped urban rasters on top of each other to make a frequency map. This can display how often and by how much certain regions of the urban areas have higher VV backscatter values than the mean, which can possibly indicate areas that are often flooded during flood. In other words, finding spatial patterns during urban floods when combining temporal VV backscatter data is the fourth recommendation.

The last recommendation is to look at the possibility of quantifying the relation between precipitation and VV backscatter peaks. Is there an average distance between the precipitation peaks and VV backscatter peaks, and is this time difference probable? Before conducting this recommendation, it is advised to change the precipitation data from local data to catchment area data, as well as improving the extent of the urban areas.

# 

# References

Belgiu, M., & Drăguţ, L. (2016). Random forest in remote sensing: A review of applications and future directions. *ISPRS journal of photogrammetry and remote sensing*, 114, 24-31.

Cohen, M. J., & Singh, B. (2014). Climate change resilience: The case of Haiti. *Oxfam International.*

Defensie (2021, August 10). Oud-CDS Middendorp: ‘Steeds groter beroep op Defensie door klimaatverandering’. Retrieved from: <https://www.defensie.nl/actueel/nieuws/2021/08/10/oud-cds-middendorp-steeds-groter-beroep-op-defensie-door-klimaatverandering>

Defensie (2022, March 7). Middendorp: “Niet alleen oorlog vormt bedreiging voor wereldvrede, klimaat ook”. Retrieved from <https://www.defensie.nl/actueel/nieuws/2022/03/07/middendorp-niet-alleen-oorlog-vormt-bedreiging-voor-wereldvrede-klimaat-ook>

Deshmukh, K. S., & Shinde, G. N. (2006). An adaptive color image segmentation. *ELCVIA: Electronic Letters on Computer Vision and Image Analysis,* 5(4), 012-23.

EO Browser (2022). EO Browser, Home, Explore. Retrieved from <https://www.sentinel-hub.com/explore/eobrowser/>

European Commission (2019). GHSL - Global Human Settlement Layer, Download the data produced by the GHSL, <https://ghsl.jrc.ec.europa.eu/download.php?ds=pop>

European Commission’s Joint Research Centre (2022). Global Surface Water - Data Access, Individual 10°x10° files, derived at 29-06-2022, from <https://global-surface-water.appspot.com/download>

Ezzine, A., Darragi, F., Rajhi, H., & Ghatassi, A. (2018). Evaluation of Sentinel-1 data for flood mapping in the upstream of Sidi Salem dam (Northern Tunisia). *Arabian Journal of Geosciences,* 11(170), 1-9, https://doi.org/10.1007/s12517-018-3505-7

Global Precipitation Measurement (2022). Ground and Airborne Instruments. Retrieved at 24-06-2022. Retrieved from <https://gpm.nasa.gov/science/ground-validation/ground-and-airborne-instruments>

ICRC. (2021). ICRC Country Profiles - Chad. International Committee of the Red Cross (ICRC). Retrieved from: <https://www.climatecentre.org/wp-content/uploads/RCCC-ICRC-Country-profiles-Chad.pdf>

Koubi, V. (2019). Climate change and conflict. *Annual Review of Political Science,* 22, 343-360.

Lacava, T., Ciancia, E., Faruolo, M., Pergola, N., Satriano, V., & Tramutoli, V. (2019). On the potential of RST-FLOOD on visible infrared imaging radiometer suite data for flooded areas detection. *Remote Sensing,* 11(5), 598.

Long, S., Fatoyinbo, T. E., & Policelli, F. (2014). Flood extent mapping for Namibia using change detection and thresholding with SAR. *Environmental Research Letters,* 9(3), 035002.

Mason, D. C., Speck, R., Devereux, B., Schumann, G. J. P., Neal, J. C., & Bates, P. D. (2009). Flood detection in urban areas using TerraSAR-X. *IEEE Transactions on Geoscience and Remote Sensing,* 48(2), 882-894.

Matgen, P., Hostache, R., Schumann, G., Pfister, L., Hoffmann, L., & Savenije, H. H. G. (2011). Towards an automated SAR-based flood monitoring system: Lessons learned from two case studies. *Physics and Chemistry of the Earth,* Parts A/B/C, 36(7-8), 241-252.

NASA POWER (2021). The Power Project. Retrieved at 24-06-2022. Retrieved from <https://power.larc.nasa.gov/>.

Noble, W. S. (2006). What is a support vector machine?. *Nature biotechnology*, 24(12), 1565-1567.

Nordås, R., & Gleditsch, N. P. (2015). Climate change and conflict. In *Competition and Conflicts on Resource Use* (pp. 21-38). Cham: Springer.

Nurmohamed, R. J., & Naipal, S. (2006). Variability of rainfall in Suriname and the relation with ENSO-SST and TA-SST. *Advances in Geosciences*, 6, 77-82.

Ontivero-Ortega, M., Lage-Castellanos, A., Valente, G., Goebel, R., & Valdes-Sosa, M. (2017). Fast Gaussian Naïve Bayes for searchlight classification analysis. *Neuroimage*, 163, 471-479.

Pal, M. (2005). Random forest classifier for remote sensing classification. *International journal of remote sensing*, 26(1), 217-222.

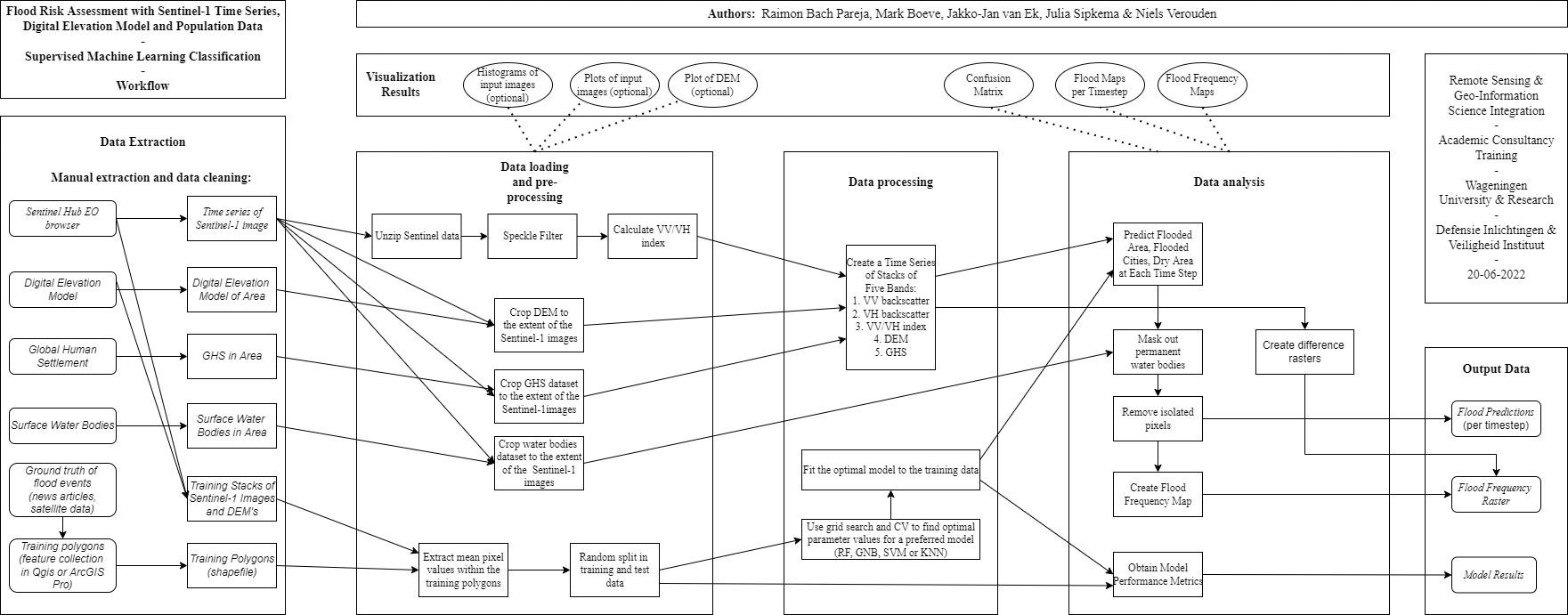
Peterson, L. E. (2009). K-nearest neighbour. *Scholarpedia*, 4(2), 1883.

Small, D. (2011). Flattening gamma: Radiometric terrain correction for SAR imagery. *IEEE Transactions on Geoscience and Remote Sensing*, *49*(8), 3081-3093.

Solaun, K., Alleng, G., Flores, A., Resomardono, C., Hess, K., & Antich, H. (2021). *State of the Climate Report: Suriname.*

# Appendices

## Appendix A - Workflows



## 

## Appendix B - Machine Learning: Grid Search Parameter Settings

Optimal parameter settings for the study area of Cap-Haitien are underscored.

Optimal parameter settings for the study area of N'djamena are marked in yellow.

| Model | Parameter | Settings | Number of Candidates |
| --- | --- | --- | --- |
| k-Nearest Neighbours | Number of neighbours | 3, 4 ,5, 6, 7 | 5 |
| Support Vector Machine | C (Regularisation parameter) | 1, 1.5, 2 | 27 |
| Kernel | Linear;  Poly;  RBF. |
| Degree | 2, 3, 4 |
| Random Forest | Minimum number of samples required to be at a leaf node | 1, 3 | 72 |
| The number of features (N) to consider when looking for the best split | 50% of N;  The square root of N;  The base-2 logarithm of N. |
| Maximum depth | 10, 15, 20 |
| Class weights | None,  Balanced,  class\_dict\_mod\*,  class\_dict\_extr\*\* |
| \*This is a custom set of class weights, calculated as the balanced set of weights, where the classes for flooded land and flooded urban are multiplied by 5.  \*\*This is a custom set of class weights, calculated as the balanced set of weights, where the classes for flooded land and flooded urban are multiplied by 10. | | | |
| More information from the Scikit Learn documentation:  [k-Nearest Neighbours](https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html), [Support Vector Machine](https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html), [Random Forest](https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html) | | | |

## 

## Appendix C - Machine Learning Results

The table below provides information on the performance of each model. The frequency maps are based on a series of 38 Sentinel-1 images taken between 01-01-2021 and 26-04-2021. In this time frame, a severe flood event occurred on 05-04-2021, which caused the river to overflow and which led to flooding in the city of Cap Haitien. The predictions for this date and a random date without flood event (01-01-2021) are shown. The prediction maps have four colours: brown for dry areas, purple for flooded urban areas, blue for flooded land and lime green for permanent water bodies.They are plotted alongside RGB images that display VV backscatter, VH backscatter and VV/VH index in red, green and blue, resp. The frequency maps show permanent water bodies in light blue and flood risk in a colour scale that ranges from green to dark red. The confusion matrices indicate the classification results of the models on the test data. Classes 0, 1 and 2 are dry areas, flooded land and flooded urban areas, resp.

Note: The prediction and frequency maps are not georeferenced and lack proper map elements. The Matplotlib library was used for the visualisation of .tiff files, loaded as arrays, rather than the Rasterio package, since the former provides more functionalities and the creation of appropriately georeferenced maps is not part of the scope of the project. However, the predictions and frequency maps are all saved as .tiff files and can be loaded into any GIS application to observe spatial patterns.

Note: A random state was used to ensure that results are reproducible. Altering the random state will affect the results.

## Cap-Haitien

| Model | Test Accuracy | User’s Accuracy | Producer’s Accuracy |
| --- | --- | --- | --- |
| **Gaussian Naive Bayes** | 0.91 | Dry: 1  Flooded Land: 1  Flooded Urban: 0.7 | Dry: 0.82  Flooded Land: 1  Flooded Urban: 1 |
| Prediction on 01-01-2021 (no flood) | | Prediction on 05-04-2021 (flood) | |
|  | |  | |
| Confusion Matrix | | Combined Flooded Frequency Map | |
|  | |  | |
| Flooded Land Frequency Map | | Flooded Urban Frequency Map | |
|  | |  | |

| Model | Test Accuracy | User’s Accuracy | Producer’s Accuracy |
| --- | --- | --- | --- |
| **k-Nearest Neighbours** | 0.94 | Dry: 1  Flooded Land: 1  Flooded Urban: 0.78 | Dry: 0.88  Flooded Land: 1  Flooded Urban: 1 |
| Prediction on 01-01-2021 (no flood) | | Prediction on 05-04-2021 (flood) | |
|  | |  | |
| Confusion Matrix | | Combined Flooded Frequency Map | |
|  | |  | |
| Flooded Land Frequency Map | | Flooded Urban Frequency Map | |
|  | |  | |

| Model | Test Accuracy | User’s Accuracy | Producer’s Accuracy |
| --- | --- | --- | --- |
| **Support Vector Machine** | 0.91 | Dry: 1  Flooded Land: 1  Flooded Urban: 0.7 | Dry: 0.83  Flooded Land: 1  Flooded Urban:1 |
| Prediction on 01-01-2021 (no flood) | | Prediction on 05-04-2021 (flood) | |
|  | |  | |
| Confusion Matrix | | Combined Flooded Frequency Map | |
|  | |  | |
| Flooded Land Frequency Map | | Flooded Urban Frequency Map | |
|  | |  | |

| Model | Test Accuracy | User’s Accuracy | Producer’s Accuracy |
| --- | --- | --- | --- |
| **Random Forest** | 0.91 | Dry: 0.94  Flooded Land: 1  Flooded Urban: 0.75 | Dry: 0.88  Flooded Land: 1  Flooded Urban: 0.86 |
| Prediction on 01-01-2021 (no flood) | | Prediction on 05-04-2021 (flood) | |
|  | |  | |
| Confusion Matrix | | Combined Flooded Frequency Map | |
|  | |  | |
| Flooded Land Frequency Map | | Flooded Urban Frequency Map | |
|  | |  | |

## N’djamena

| Model | Test Accuracy | User’s Accuracy | Producer’s Accuracy |
| --- | --- | --- | --- |
| **Gaussian Naive Bayes** | 0.72 | Dry: 0.67  Flooded Land: 1  Flooded Urban: 0.33 | Dry: 0.91  Flooded Land: 0.78  Flooded Urban: 0.2 |
| Prediction on 15-09-2020 (no flood) | | Prediction on 02-11-2020 (flood) | |
|  | |  | |
| Confusion Matrix | | Combined Flooded Frequency Map | |
|  | |  | |
| Flooded Land Frequency Map | | Flooded Urban Frequency Map | |
|  | |  | |

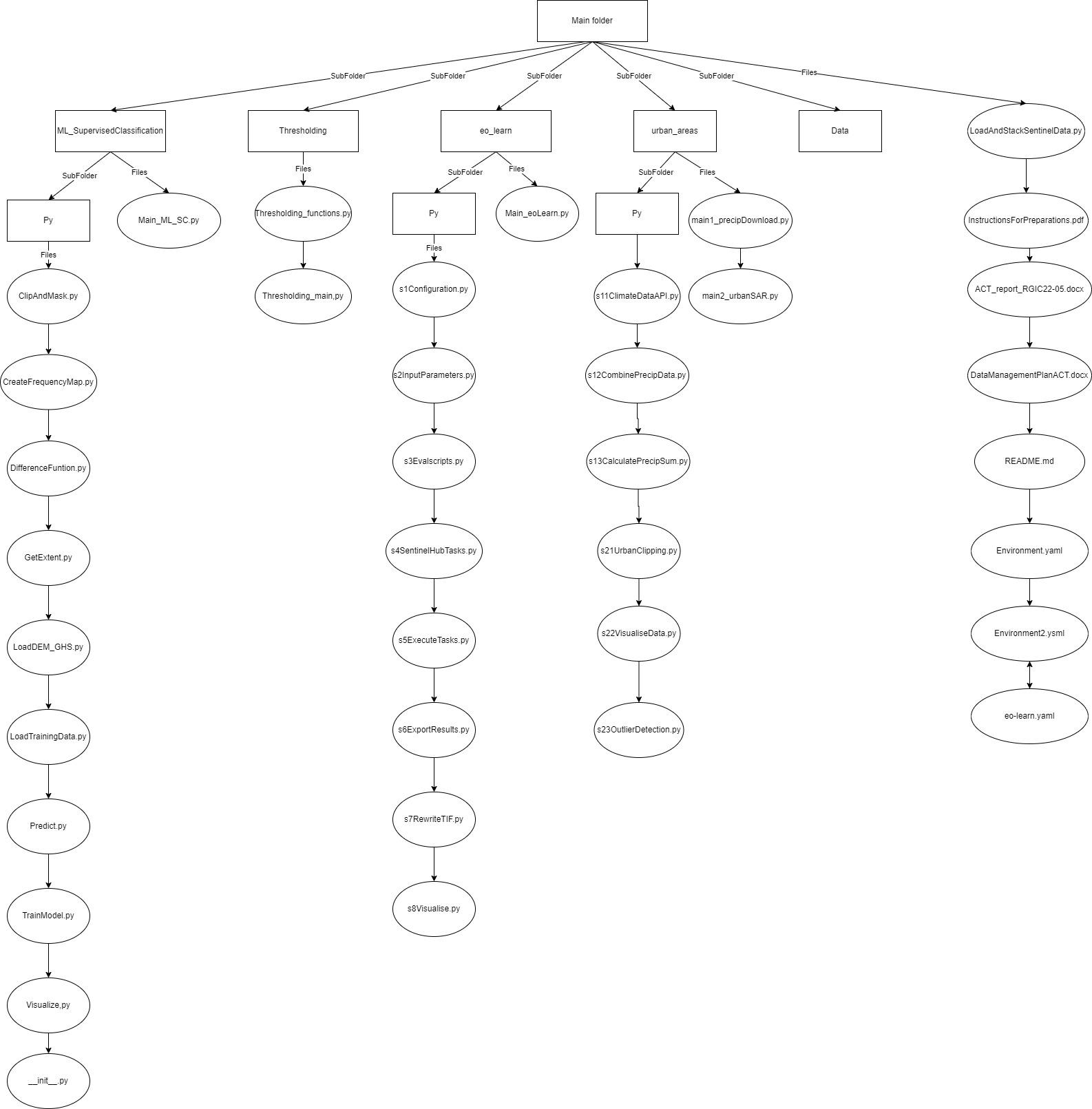
| Model | Test Accuracy | User’s Accuracy | Producer’s Accuracy |
| --- | --- | --- | --- |
| **k-Nearest Neighbours** | 0.84 | Dry: 0.77  Flooded Land: 1  Flooded Urban: 0.67 | Dry: 0.91  Flooded Land: 1  Flooded Urban: 0.4 |
| Prediction on 15-09-2020 (no flood) | | Prediction on 02-11-2020 (flood) | |
|  | |  | |
| Confusion Matrix | | Combined Flooded Frequency Map | |
|  | |  | |
| Flooded Land Frequency Map | | Flooded Urban Frequency Map | |
|  | |  | |

| Model | Test Accuracy | User’s Accuracy | Producer’s Accuracy |
| --- | --- | --- | --- |
| **Support Vector Machine** | 0.76 | Dry: 0.69  Flooded Land: 1  Flooded Urban: 0.33 | Dry: 0.82  Flooded Land: 1  Flooded Urban: 0.2 |
| Prediction on 15-09-2020 (no flood) | | Prediction on 02-11-2020 (flood) | |
|  | |  | |
| Confusion Matrix | | Combined Flooded Frequency Map | |
|  | |  | |
| Flooded Land Frequency Map | | Flooded Urban Frequency Map | |
|  | |  | |

| Model | Test Accuracy | User’s Accuracy | Producer’s Accuracy |
| --- | --- | --- | --- |
| **Random Forest** | 0.80 | Dry: 0.71  Flooded Land: 1  Flooded Urban: 0.5 | Dry: 0.91  Flooded Land: 1  Flooded Urban: 0.2 |
| Prediction on 15-09-2020 (no flood) | | Prediction on 02-11-2020 (flood) | |
|  | |  | |
| Confusion Matrix | | Combined Flooded Frequency Map | |
|  | |  | |
| Flooded Land Frequency Map | | Flooded Urban Frequency Map | |
|  | |  | |

## Appendix D - Overview of files

In this project many files and folders have been created to order all our data, scripts and reporting. This appendix will give a short overview of all these folders and files and what they are for. Below, an image visualising the data is shown (zooming in might be necessary). Afterwards a short explanation of the folders and files is given.



Folders:

* ML\_SupervisedClassification:

Here all the files relating to the machine learning algorithms are stored. In the main.py the whole script can be run to get the main machine learning results. The script calls all the different functions stored in the py folder.

* Thresholding

In this folder there are two files to run the thresholding method: the main thresholding script and the functions. As with the classification only the main needs to be run, since the functions of the other script are called here.

* eo\_learn

In this folder a script to download sentinel data can be found. The main file is the API\_main.py. This script runs all the functions made in the different files in the py folder.

* Urban\_areas

This folder contains the scripts for both downloading precipitation data with the POWER API and the urban flood analysis. The main script for the former is called “main1\_precipDownload.py” and the main script for the urban flood analysis “main2\_urbanSAR.py”. The different functions that correspond to these two “main” scripts are given in the “py” folder.

Files:

* InstructionsForRunningTheTimeSeriesApplication.docx

This file describes how the data can be downloaded and should be stored for the models to work well.

* README.md

In the README file a short overview of the project is written in order to understand what the code does.

* Report

The report, this file, describes the background, the methods, the results and the conclusions of the project. It also shows the discussion and recommendations for the future.

* DataManagementPlan.docx

In the data management a description is given of what data is used in the project and how it is stored and will be stored in the future.

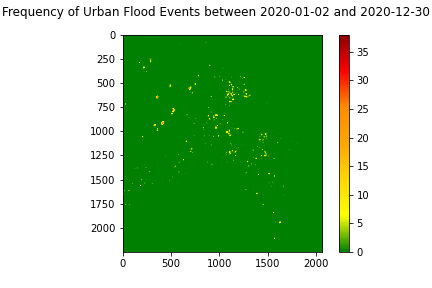
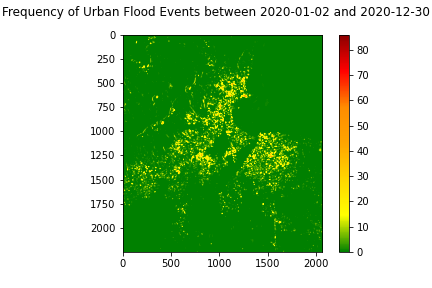
* Environment.yml, environment2.yaml, eo-learn.yaml

The .yaml files contain all the packages that have to be installed (many others are in the standard python library already). Note: if the Sentinel Hub package is not used, the eo-learn.yaml file can be disregarded.

## 

## 

## Appendix E - Additional result of image differencing



*Flood frequency maps of urban areas which is created by image differencing of SAR imagery of the year 2020. Left: threshold = 0.4, right: threshold = 0.5*

From the year 2020, there were 100 images used as input for the model which took about half an hour to run. The extent is different from the extent of the result which used 6 images and it is from a different year. According to the result with a threshold of 0.5, a lot of parts of the city of Cap Haitien has been flooded between 10 and 20 times in the year 2020, which did not really happen. The results prove that this method is not fully developed yet and it is hard to do so, in the discussion the validity of the results are discussed more elaborately.