Heidelberg University

Fundamentals of Machine Learning

Detecting Volcanoes on Venus using the Magellan Imageset

Maximilian Klingmann, Julien Stern

supervised by
PD Dr. Ullrich Köthe

Matriculation Number: 2932908, 3057886
Date of Submission: 2018/03/22

Declaration of Authority

We, **Maximilian Klingmann, Julien Stern**, hereby declare that this essay, titled **Detecting Volcanoes on Venus using the Magellan Imageset**, written for the lecture **Fundamentals of Machine Learning** held in **WT 17/18** and supervised by **PD Dr. Ullrich Köthe**, and all material presented were created by ourselves entirely. Furthermore we declare that all adaptions from other sources are specifically acknowledged throughout the report. Citation, as well as the usage of foreign sources, texts or any other aid are denoted according strict scientific rules. We understand that we must not present foreign texts or text passages as work of our own. Doing so is seen as cheating and violates basic rules of scientific work. We are aware that cheating would lead to a withdrawal of the examination among other consequences. We declare that all statements and information contained herein are true, correct and accurate to the best of our knowledge and belief.

Leimen, 2018/03/22

Abstract

The surveillance and examination of Venus as so-called Ëarth's sisterẗhas been subject to research for over three decades now, including the launch of the *Magellan* mission in 1989. The goal of the mission was a complete mapping of the Venus surface with a synthetic-aperture radar. It produced around 30000 images of the Venus' surface. However annotating these images by hand is a time-consuming matter and has been done for around 130 images by human experts. The poor quality of the images make it even harder to annotate potential volcanoes. To aid in these annotation tasks a Machine Learning approach for automated labelling of potential volcanoes is employed. This report uses a subset of the *Magellan Imageset* and takes up on this challenge. Various Machine Learning classifiers like K-nearest Neighbour, Linear Discriminant Analysis and a Neural Network are trained to discover potential volcanoes using a Sliding Window approach on the images provided. The Ground Truth given by the human experts is divided into four classes ranging from *"Certainly a volcano"* ($p >= 0.98$)to *"Maybe a volcano"* ($p >= 0.5$). The afore mentioned methods to classify are not able to distinguish between all four classes and another class for *Certainly not a volcano*, which covers any area not covered by the former and do perform poorly (accuracy 0.3). However by combining classes together the methods are able to find volcanoes a lot better. When combining the four classes used by the human experts into one class and having the other class stay another class the methods are able to score an accuracy of around 0.8.

Contents

1 Introduction

In the first chapter of this report we introduce the topic and state the motivation behind detecting volcanoes on Venus with Machine Learning methods. We then proceed to the explanation about the used dataset used during the project and eventually explain the used terminology used throughout the report.

1.1 Motivation

Because of the many similarities between Venus and Earth like size, mass, density and volume the planet is often referred to as *Earth's sister*. It is also assumed that both plan mets share common origin and formed approximately at the same time around 4.5 billion years ago. Furthermore Venus was the planet on which the runaway greenhouse effect was first discorvered and the planet itself is surrounded by a much more dangerous and un-friendly atmosphere than Earth is. The atmosphere is compose of carbon dioxide, which generates temperatures at the surface hot enough to melt lead and a surface pressure that is 90 times the pressure on Earth [1]. These information taken from the NASA website alone makes Venus an interesing planet to learn from and might make it possible to transfer gained knowledge to the Earth's future being.

Hence the exploration and surveying of Venus has been subject to research for many years now and in 1989 the *Magellan* spacecraft was launched to map the surface of Venus. It took around 30000 images in the years 1991 to 1994 with a so-called synthetic-aperture radar. The format of the images is

[1] https://www.nasa.gov/topics/solarsystem/features/magellan20100408.html

explained later on in this report. Following the capturing of the images Burl et al. [2] first tried recognize volcanoes on Venus with classification methods but had to deal with the fact that the images taken were annotated by human experts to generate semi-valid *Ground Truth*. This is further discussed in the chapter called D̈ealing with uncertain Ground Truthẅithin this report.

The methods applied in this project expand the ones used by Burl et al. by new classifiers. However opposing to Burl et al. who used the complete dataset, we were only able to learn our classifiers on roughly 130 images because the full dataset is not easily available any longer. We first employed a dimensionality reduction using *Principal Component Analysis (PCA)* to narrow down overfitting issues and learned various classifiers after the PCA transformation including *Linear Discriminant Analysis (LDA)*, *Quadratic Disctriminant Analysis (QDA)*, *K-nearest Neighbour (KNN)* and a *Neural Network (NN)*.

1.2 Dataset

In the following section we describe the dataset we used, how the images were created and what the images show as well as the provided pre-processed images.

1.2.1 The Magellan Imageset

The *Magellan* spacecraft launched on May 4, 1989 by NASA also referred to as the *Venus Radar Mapper* was used for the *Magellan* mission, which goal it was to map the surface of Venus using *Synthetic-aperture radar (SAR)*. This method allows to create two-dimenstional images of landscapes. The produced greyscale images range from values between 0 and 255, where lower values indicate low energies backscattered to the radar and higher values indicate greater backscatter respectively. This makes it possible to locate elevations of the surface of the planet. Figure 1.1 illustrates this. One can see that the radar picks up greater backscatter on the elevated side of a hill or mountain which faces the radar, while picking up lesser backscatter on the opposing hill side.

During the mission the spacecraft produced approximately 30000 images throughout the years 1990-1994. However in the dataset provided only 134 images are included. All images are 8-bit grey-scaled and have a resolution 1024x1024 pixels spanning 30 km^2 on the planets surface. Each image is assisted by a file containing the "Ground Truth" of the image containing coordinates and radii of volcanoes and a corresponding label. The labels are described below.

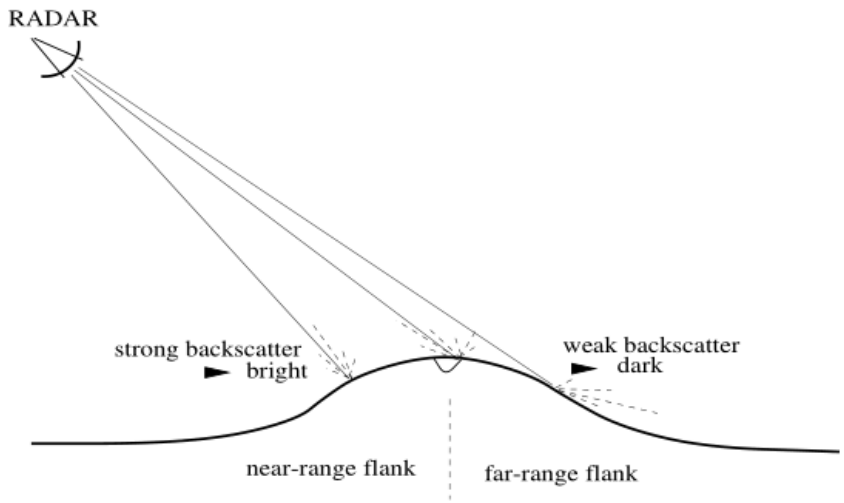


Figure 1.1: This image shows how the radar images are created. Elevations facing the radar strongly backscatter and lead to higher values in the resulting image, while elevations facing away have weak backscatter and are therefore darker in the resulting image

1.2.2 Classes

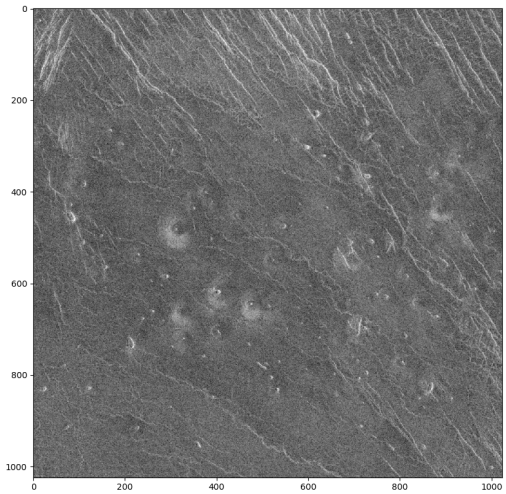
The *Ground Truth* given in the dataset is not real *Ground Truth* as no one has been to Venus yet. Instead the potential volcanoes have been assigned to a class by human experts. As described before each image comes with a file listing coordinates and radii of potential volcanoes and assigning each volcano a class. Each class is represented by a label. The labels range from 1 to 4. The meaning of each of the labels is given in List 1.2 below.

Figure 1.2a and figure 1.2b show two images taken by the *Magellan* spacecraft. One can clearly see the dents in the first image which indicate potential volcanoes. However looking at the second image one can hardly recognize any volcanoes and thus even the human experts do not fully agree on certain areas. This problem and how we tackled it is mentioned again in the Feature Extraction section.

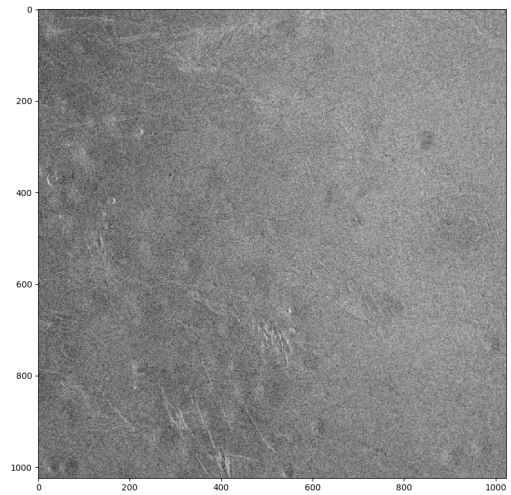
- Label 1: "Almost certainly a volcano" ($p = 0.98$) - "the image clearly shows a summit pit, a bright-dark pair, and a circular planimetric outline"
- Label 2: "Probably a volcano" ($p = 0.80$) - "the image shows only two of the three category 1 characteristics"
- Label 3: "Possibly a volcano" ($p = 0.60$) - "the image shows evidence of bright-dark flanks or a circular outline; summit pit may or may not be visible"
- Label 4: "Maybe a volcano" ($p = 0.50$) - "the image shows a visible pit but does not provide conclusive evidence for flanks or a circular outline.

List 1.2: The four labels used in the dataset to distinguish different kinds of potential volcanoes.

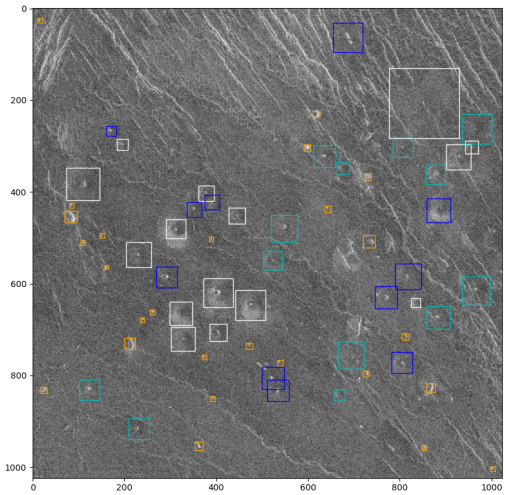
Disregarding the fact that the human experts do sometimes not agree on labels for potential the dataset does offer the coordinates and radii accompanied by the labels as mentioned above. Figure 1.3a and figure 1.3b show the same images and all the volcanoes mentioned in the *Ground Truth* for the images. The size of the rectangles are chosen to be twice the radii of the volcanoes they enclose. The rectangles' colors depict the labels the volcanoes within have been assigned to, whereas white indicates Label 1, blue indicates Label 2, cyan indicates Label 3 and orange indicates Label 4. Again one can see that there are potential volcanoes which can be hardly seen by the human eye, especially areas that belong to Label 4.



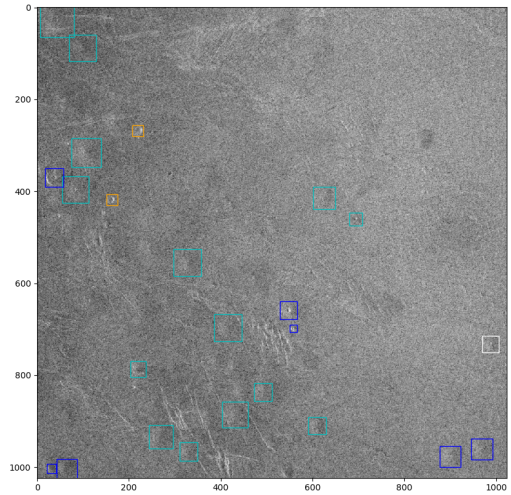
(a) Image 3 of the dataset. The potential volcanoes are clearly visible in the lower left area of the image



(b) Image 66 of the dataset. In this area the volcanoes are hardly visible with the human eye



(a) Image 3 with all volcanoes as mentioned in the Ground Truth



(b) Image 66 with all volcanoes as mentioned in the Ground Truth

1.3 Terminology

Throughout this report there are several terms repeatedly used which might need to be explained beforehand due to their nature of use. A list of acronyms can be found at the very end of this report.

1.3.1 Ground Truth

In order for a machine learning method to learn classes a dataset with so-called *Ground Truth (GT)* is needed. The GT gives information on category assignments for each feature vector. In Image Processing for example this can be a dataset consisting of various animals, where each image is assigned a label like "Cat", "Dog", "Giraffe". This information is vital for the learning method. When feeding these images and their corresponding GT to the method it tries to learn distinguishable features in the various categories. The learning is done on a fraction of the whole dataset, e.g. 80% while the the rest is used as a test set. After the method learned the labels from the training set it is told to predict the labels of the test set and returns a set of labels, one for each image given in the test set. Since the GT of the test set is also known one can now compare the "real labels" of the test set images with the predicted labels and evaluate the quality of the method.

1.3.2 Precision & Recall

Precision & Recall are used to evaluate the results of an algorithm. The precision shows the ratio between the items returned as positive matches against the items expected when considering the Ground Truth. It is defined as:

$$p = \frac{t_p}{t_p + f_p} \tag{1.1}$$

where t_p depicts the *True positives* which are the relevant results and f_p ,the *False positives* or the fraction of results that are irrelevant. The recall

measures how many of the relevant items have been selected by the algorithm as positive matches:

$$r = \frac{t_p}{t_p + f_n} \tag{1.2}$$

where t_p depicts the above and f_n depicts the *False negatives* or the items falsely accused to be not positive.

1.3.3 F1-Score

The F1-Score gives the harmonic average of the Precision & Recall. It is computed the following way

$$F_1 = 2 \cdot \frac{precision \cdot recall}{precision + recall} \tag{1.3}$$

1.3.4 Accuracy

Accuracy is the number of matching labels between prediction and the labels which should have been predicted according to Ground Truth in the tested dataset in ratio to the number of all labels. Example: If the method predicted 390 labels for a total of 400 images correctly it has an accuracy of 0.975.

1.3.5 Overfitting & Underfitting

Overfitting/Underfitting occurs in Machine Learning when the feature vectors are poorly chosen. Overfitting happens whenever too much information of a feature vector are taken into account making the fitting of the classifier working well on the training set but performing poorly on the test set, e.g. Considering too many pixels in an image as important for a specific class. Underfitting is the opposing phenomenon, meaning too many features are excluded because of apparent irrelevance for a given class, when in reality some features might be very representative for that given class.

2 Methods

In this chapter we present the methods we used to classify the volcanoes. First we briefly describe the *Dimensionality Reduction* and what *Classifiers* we chose to classify the images. We then proceed to the bigger section in this chapter describing our *Feature Extraction* and the challenges we had to tackle.

2.1 Dimensionality Reduction

To avoid overfitting by taking each pixel of the 1024x1024 images into account when learning the various classifiers we decided to utilize a *Principial Component Analysis (PCA)* to narrow down each image to pixels only. This allowed us to also plot the PCA and thus manually verify the quality of the fitting. Figure 2.2 (left) shows one instance of transformed training data after the PCA was fitted to it. For this example all four labels were untouched and thus taken as they were provided in the dataset. Also shown is that the PCA is troubled by the noise of the images, i.e. the stripes seen at the top of Figure 1.3a, and also by potential volcanoes looking similar but having assigned different labels in the training set. This leads to the cluttered image. The data is not linearly separable and thus the decision boundaries set by the PCA do not reflect the classes well.

2.2 Feature Extraction

In the PCA mentioned before the feature vectors were so-called *Chips* provided by the dataset. The chips are pre-processed smaller areas of the images which contain one volcano. Each chip thus is assigned a label ranging from 0-4.

The newly added label 0 indicates a chip where no volcano is present. The chips provided are specifically designed for Machine Learning purposes and to make it easier to start with the ML approaches. That is the reason we decided to use these chips at the start of our project. However these chips are highly unbalanced, where the amount of chips labelled in Label 1 is about 150 and the amount of chips labelled in Label 0 is approximately 600. We tried balancing the set by only allowing a certain amount of chips for each label, but that did not improve the PCA, since it forced use to ignore many features.

2.3 Patches

We tackled this problem by moving on to the provided Ground Truth which contains labelled locations, given in x/y coordinates and corresponding radius of a volcano candidate on a given image. We decided to extract the features from scratch by only using the images and the Ground Truth given in the dataset. Iterating over all images and their accompanied Ground Truth file we extracted 32x32 pixel patches, which were centered at the volcanoes coordinates. Each patch was assigned the label given in the dataset for the corresponding coordinate. The patch size was chosen to allow the average radius of a 1,2 or 3 labelled feature to fit completely into one patch centered at the volcano center. The average radius of features labelled with 4 is way smaller than the rest of the set, so we neglected radii of label 4 features. Figure 2.1 shows example patch drawings of the four categories 1-4. An classification approach at this state of feature extraction is depicted in Figure 2.2. The corresponding confusion matrix is shown in Figure 2.3(left). Furthermore we extracted patches at randomly chosen coordinates among all images with the constraint of not being close to a Ground Truth specified volcano. These patches were assigned label 0. Figure 2.4 and Figure 2.3(right) show the results. Allthough adding these zero patches did not bring significant performance

gain for the testing on patches, it was necessary to test the results for the sliding window approach. This behavior is present over all classifier (LDA, QDA, KNN and NN), which is why we only show it using the KNN as an example.

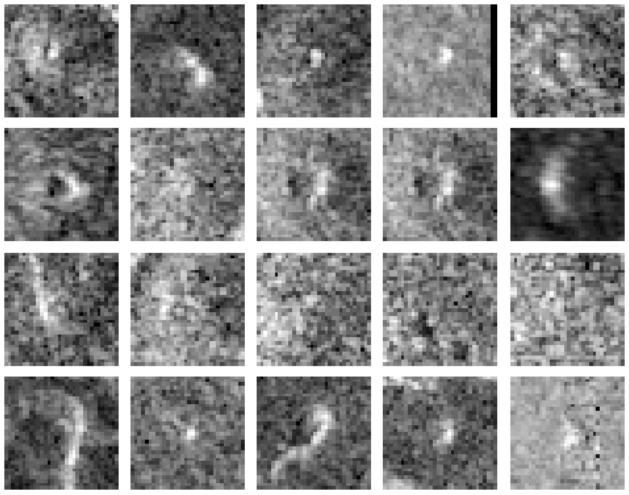


Figure 2.1: Randomly chosen patches for the classes 1-4 respectively. The row number corresponds to the classes the volcanoes within the patches are assigned to, hence row one shows volcanoes with label 1.

2.3.1 Dealing with uncertain Ground Truth

The dataset we used in this work does not contain real Ground Truth, but labels assigned by human experts (planetary geologists). No one has ever been to Venus and since that no conformation exists, of whether or not a spot on an image taken from the Magellan spacecraft is actually a volcano. Even among domain experts who were responsible for labelling the present dataset, there exists a certain disagreement on assigned labels. Given that, both the human labelling and the machine learning process we employed, underlay an uncertainty problem regarding the provided Ground Truth. To illustrate these circumstances Burl et al. [2] describe an experiments, where two experts' (A

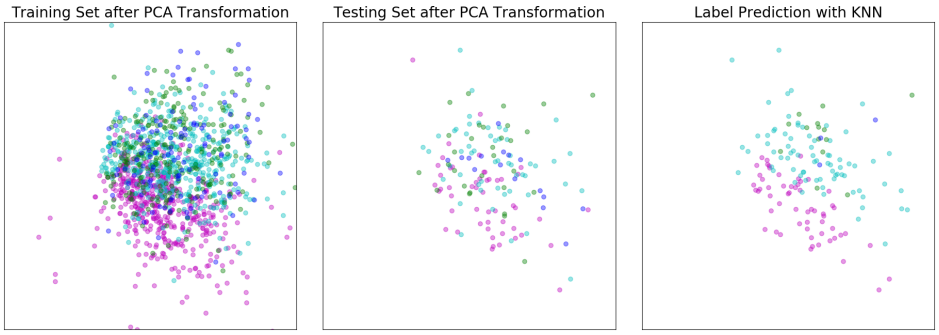


Figure 2.2: First image shows the transformed training set after the PCA has been fitted to it. The big cluttered area shows that the PCA has trouble separating the four labels therefore finding correct decision boundaries are impossible when using the out-of-the-box labels provided by the dataset. The image in the middle shows the test data labelled according to Ground Truth and the right most image shows the prediction of the KNN classifier. Labels are represented in the following colors: label 1 = blue, label 2 = green, label 3 = cyan, label 4 = magenta.

and B) opinions on a subset of images from the original dataset are compared. If two visual features (one from expert A and one from expert B) are within a few pixels to each other, they are considered to be the same visual feature. This tolerance is necessary to allow for labelling comparison. The result is given in form of a confusion matrix ([2] Table 1), showing relatively high numbers in non-diagonal cells, indicating the aforementioned uncertainty problem. However the matrix also shows higher consensus on features labelled as class 1 and 2, as on those labelled as 3 and 4, suggesting feature merging as described in the following paragraph.

We decided to merge features labelled as class 1, 2 and 3 into one common feature class, furthermore considering features in this class to have label 1. This step significantly increases the algorithm's performance on finding features that have at least a 60% chance to count as a volcano. Burl et al. [2] even consider features from all class to be "true" volcanoes to increase algorithmic performance. They also state that from a geologist standpoint

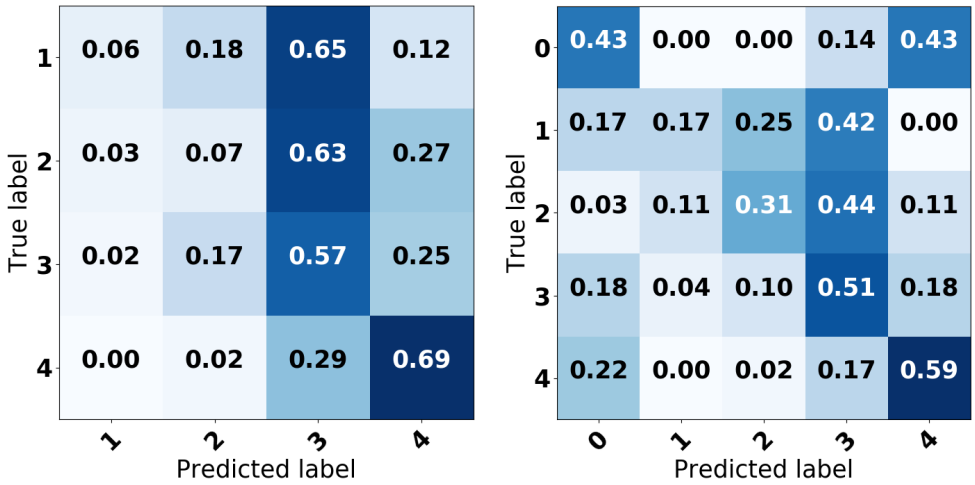


Figure 2.3: The confusion matrices for the PCA + KNN classification approach using the labels provided by the dataset (left) and additional zero patches (right).

detecting volcanoes of class However our experiments led us to only merge 1,2 and 3 providing the best results in both precision and recall. Although Smyth et al. [3] state that for humans labelling volcanoes into quantized probability bins lead to a more effective approach than forcing "yes/no" decisions, since it "allows more accurate estimation of the underlying discriminant surfaces", providing the same bins to our learning algorithm results in worse performance in the testing phase. That is, provided with a homogeneous set of training and test images (only patches of volcano candidates labelled with 1,2,3 or 4) the classifiers score 0.4 to 0.55 F1-scores which by itself is way to low in order to label unseen data effectively. Given this performance, the task of locating volcanoes on an 1024*1024 image with a sliding window approach using 32*32 as the window size, becomes even more impractical. This is caused by the huge amount of 0-windows (a window where no volcano candidate is present at all) provided to the predictor in form of the sliding window. Experiments showed that with multi-class learning algorithms the sliding window approach produces to many false-positive locations, resulting in an image cluttered

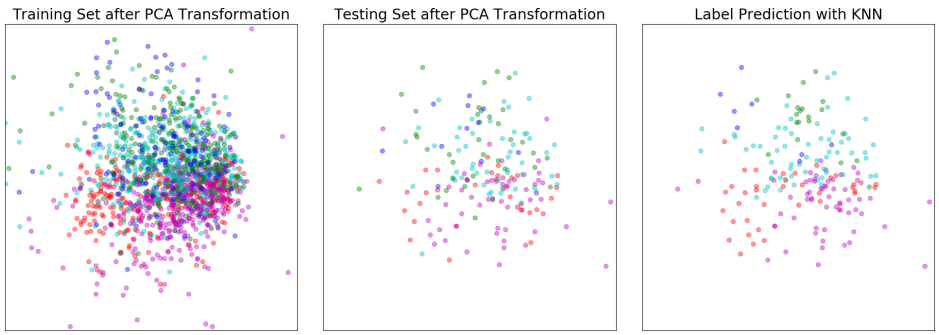


Figure 2.4: Same Figures as in Figure 2.2 but with additional zero patches displayed in red.

with supposed volcanoes.

Illustrating the performance gain, Figure 2.5 shows the same PCA + KNN approach as in Figure 2.4, but with one class containing features originally labelled as 1,2 and 3, and another class containing features labelled as 4 and 0. The corresponding confusion matrix is shown in Figure 2.6. As can be seen in both images, the combination of features lead to a better separation performance for the KNN classifier. Furthermore we will refer to this data representation as *combined dataset/labels*. This modification also dramatically increases the performance of the classifiers when faced with many 0-class windows originating from a sliding window approach.

2.3.2 Sliding Window Approach

We employed a sliding window approach on the images from the original dataset. Each window is a 32x32 pixel cutout section of an image and therefore has the same dimensions as the patches extracted for training and testing. With each iteration we advanced by half the windows size, leading to overlap of windows, e.g. if a window starts at pixel $(0, 0)$ the next window starts at pixel $(16, 0)$. We used this advance technique in both x and y direction. For each image we predicted its label and draw a rectangle on the current

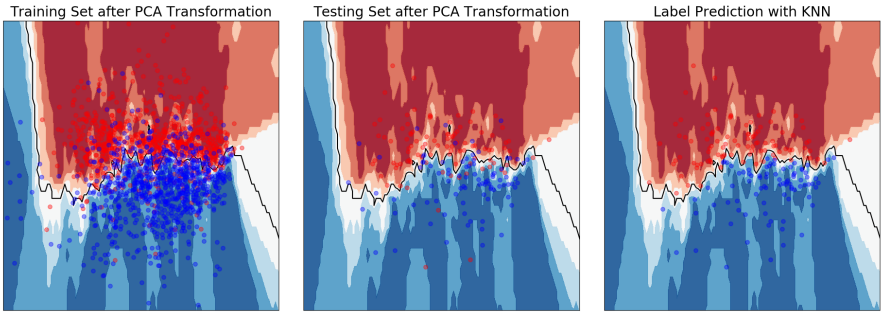


Figure 2.5: These images first show the PCA transformation of the training and test data and then the prediction of the labels for the test data of the KNN with combined labels. The black lines show the decision boundaries.

windows center pixel if the classifier decided to label it with 1.

2.3.3 Image Pre-Processing

Pre-processing the images before splitting them up into patches was subject to discussion during our project. The images themselves are of poor quality and thus the method might not work well because of it. We tested various pre-processing steps implemented in skimage, i.e. Histogram Equalization, Histogram of Oriented Gradients, Gaussian Blurring, Erosion/Dilation, Opening/Closing. However all applied methods did not improve our images by a significant margain, which is why we will not go into further detail about their functionalities.

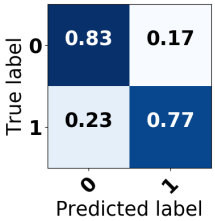


Figure 2.6: Confusion Matrix of predicted test labels versus the Ground Truth labels after a PCA transformation and an KNN prediction with combined labels.

2.4 K-Fold Cross-Validation

When dealing with a small amount of Ground Truth, like it is in our case with only 134 images being provided by the dataset the classifying method is prone to overfit on the training data. This can be avoided by using the K-Fold Cross-Validation. The method divides the dataset into k different splits. Now running the learning and testing method k times each split is used as test set once, while the k-1 other splits are used for learning. In the end we computed F-Score, Accuracy, Precision and Recall by averaging them over all iterations respectively.

2.4.1 Voting Approach

Since we ran each method k times because of the K-Fold Cross-Validation we decided to add a voting approach, where a potential candidate is assigned a class label only considered if in more than eight out of the ten iterations the method used decided the candidate belonged to that class. This eliminated candidates where the prediction was very unstable and too dependent on the chosen training/test configuration.

3 Evaluation

In this chapter we talk about the various experiments we performed and compare them. First we describe the results from a *Linear discriminant analysis* (LDA), *Quadratic discriminant analysis* (QDA) and an *Neural Network* (NN) classifier on the combined labels set. Second, we evaluate their performance when applied to the sliding window approach for two given images.

3.1 Training and Testing on the combined dataset

In this section the four aforementioned classifiers' results on the combined dataset are described. For all classifiers we used the standard settings defined by the sklearn framework [4]. The first row of Figure 3.1 shows the result of a LDA classifier. Since only two classes remain after the combination of features, the LDA performs way better than on the original 4/5-classes dataset. The confusion matrix for the given classification is shown in Figure 3.2. The second and third row of Figure 3.1 show results from the QDA and NN classifiers, respectively. All three classifiers achieve diagonal entries ≥ 0.77 in their confusion matrices which is a more promising result for further sliding window testing, as the classification on the original dataset or the dataset with append zero-patches.

3.2 Results for the sliding window approach

Before testing the classifiers with the sliding window approach we created 10 classifier instances for each classifier type using the K-Fold Cross-Validation

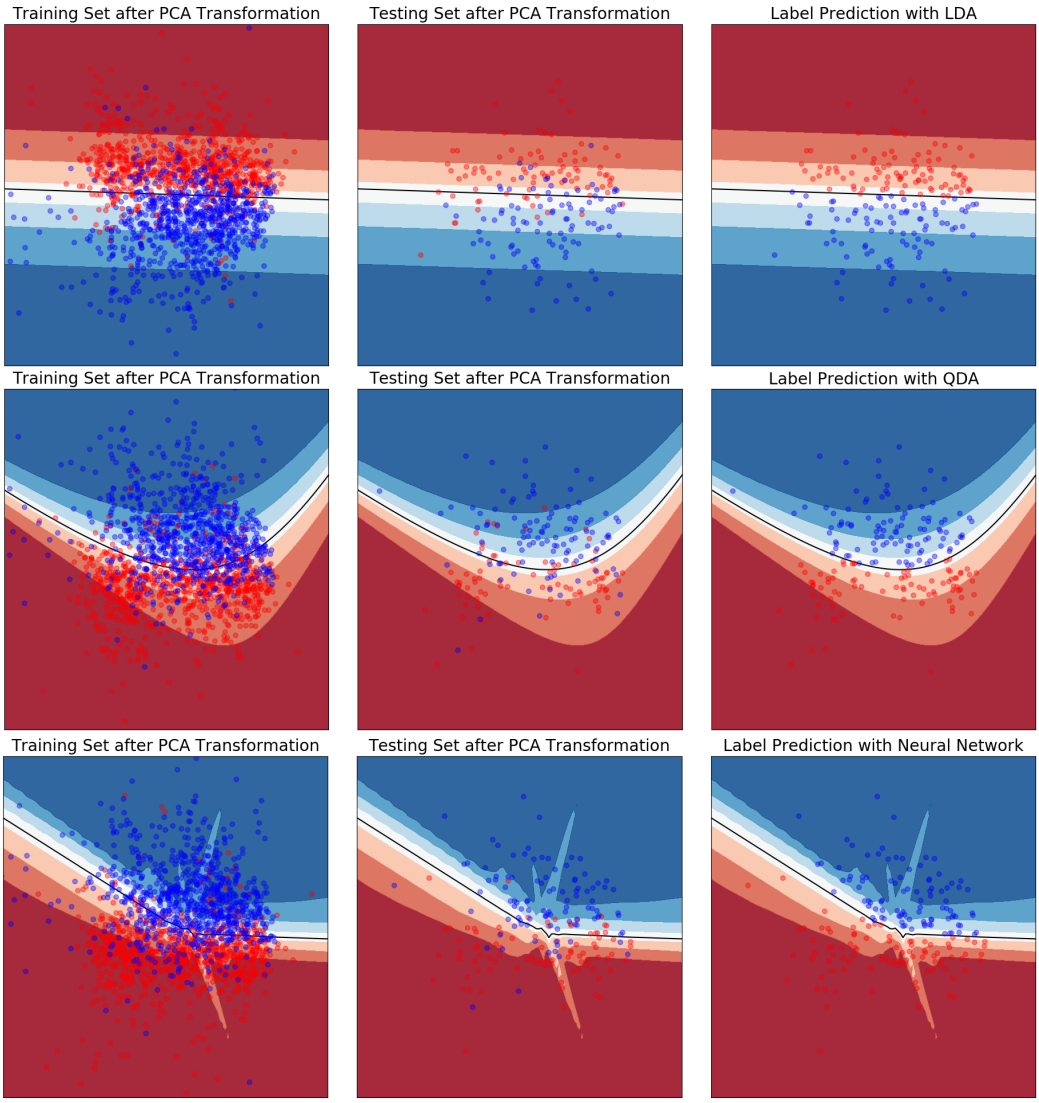


Figure 3.1: Results from the three classifiers LDA (first row), QDA (second row) and NN (third row) on the combined dataset. Red/Blue points represent label 0/1.

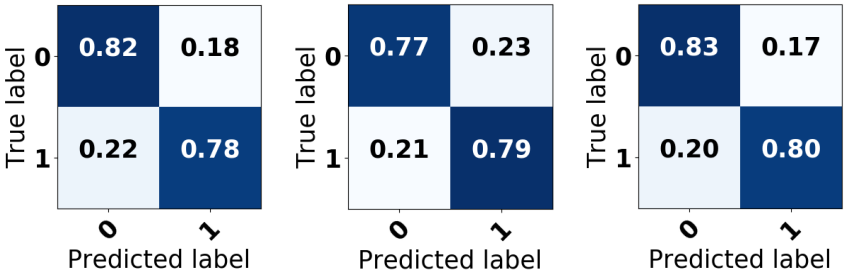


Figure 3.2: Confusion matrices for the three classifiers LDA (left), QDA (middle), NN (right).

approach, resulting in a total of 40 instances. Hence, each instance is trained and tested with a random sub-set provided by the K-Fold method. In all cases the voting threshold was set to 0.8, so 8 out of 10 classifier instances had to label a feature with 1, in order for this feature to finally be labelled with 1 and to trigger drawing a rectangle. We tested the this set of classifiers on two given images: 3 and 66. We choose these images, since they contain fundamental observations we made during testing. Firstly, as can be seen in Figure 3.3 in the upper right quarter of the image, all classifiers show large numbers of false positives on rough terrain. This is probably caused by the fact that high back scattering areas tend to confuse the classifiers because they are trained to recognize volcanoes that have this high scattering effect usually in the middle of the patch; in the pit. Especially the QDA falls for this kind of terrain type. The implementation of the classification process underlying this work allows to pick an image randomly and perform the sliding window approach on this image - the described phenomena appear on all of them.

Figure 3.4 was chosen as the counter part in terms of terrain properties. The hole image is relatively flat and only in the lower left quarter there are some mountain chains causing false positives. However almost all classifiers recognize all volcanoes with at least one window. The only volcano candidate missed by all classifiers is the label 3 feature located at approximately

$(700, 480)$ in (x, y) pixel coordinates. Only KNN missed the label 3 feature at approximately$(650, 400)$. Both features however are originally labelled as class 3 candidates, which is already very vague (60%) given the feature is a candidate.

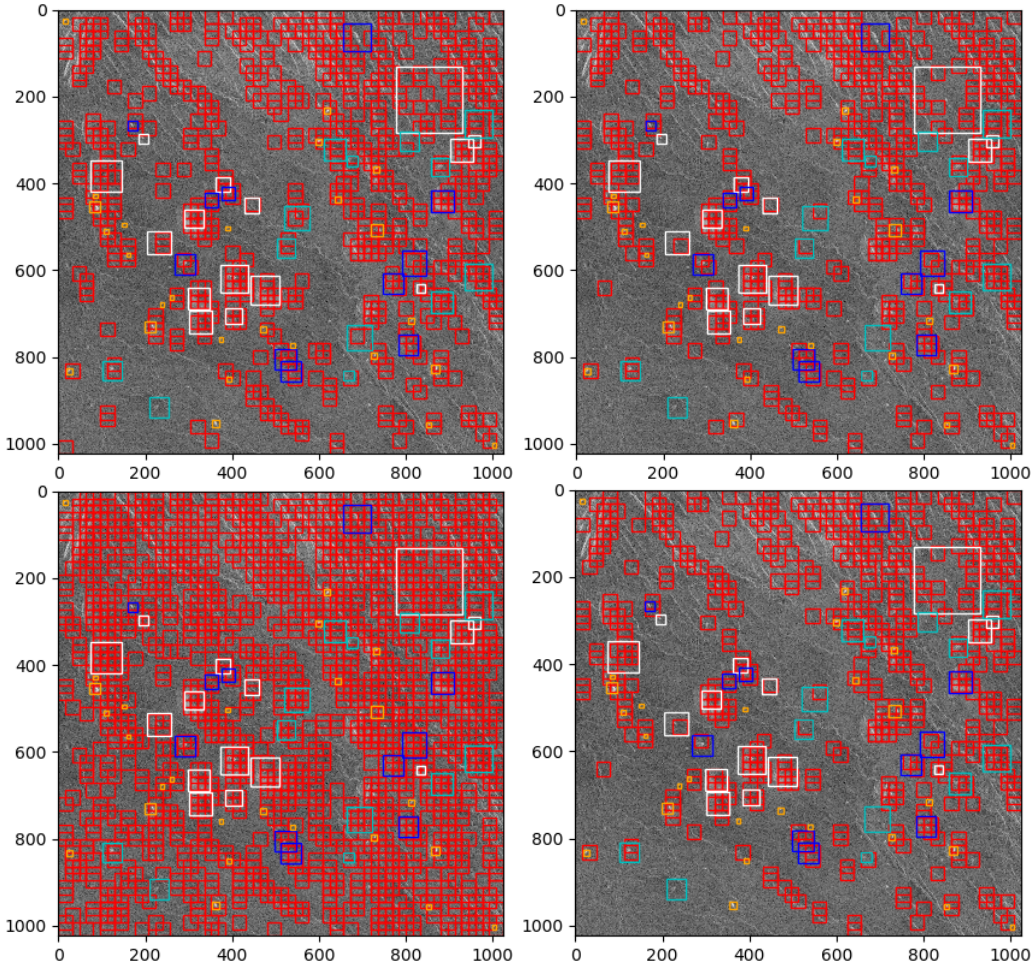


Figure 3.3: Sliding window results using image 3. From left to right/top to bottom: KNN, LDA, QDA, NN. Color coding: label 1 = white, label 2 = blue, label 3 = cyan, label 4 = orange. Note that for reference response we colored the rectangles using their original labels, although the classifiers where trained with the combined dataset.

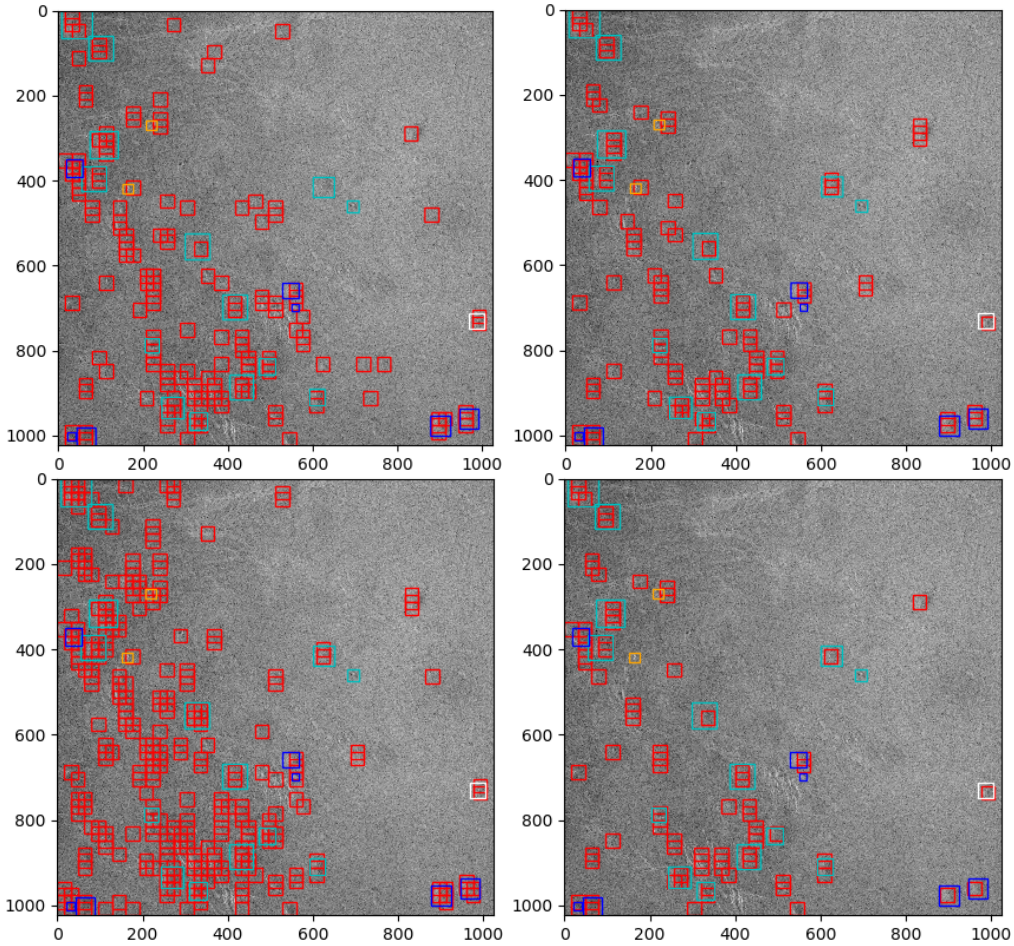


Figure 3.4: Sliding window results using image 66. From left to right/top to bottom: KNN, LDA, QDA, NN. Color coding analogous to Figure 3.3.

4 Conclusion and Future Work

As seen in the evaluation of the four classifiers, trained on different modification of the provided dataset and tested with a K-Fold validation voting system and a sliding window approach, combining features is the main improvement in terms of algorithmic performance among all classifiers. Providing the raw dataset to the PCA leads to bad separability, since the given Ground Truth is highly uncertain. Compared to the labelling of the human expert assemble, the classifiers proved to produce reasonable classification; given the average accuracy, precision and recall for all classifiers with values around 0.8 and small variances (≤ 0.05). However all tests with high back scattering terrain produces high false positive rates in these areas. We consider a pre-processing step, where the presence of such features is reduced in the image before learning even begins. Another way to tackle this issues could be to feed an increased amount of patches into the training phase, which contain these critical areas.

Another approach to increase performance is to consider altering the patch sizes in order to fit the various radii of volcano candidates. This might result in better classification, since all patches would contain the hole volcano and therefore eliminating the risk of losing the typical planimetric outlines.

Bibliography

[1] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features, 2001.

[2] Michael C. Burl, Padhraic Smytht, Usama M. Fayyadt, Pietro Peronri, Larry Crumpler, Jayne Aubele, and Lars Asker. Learning to recognize volcanoes on venus, 1998.

[3] Padhraie Smytht, Michael C. Burl Is, T Jet Propulsion Laboratory, and Usama M. Fayyad. Knowledge discovery in large image databases: Dealing with uncertainties in ground truth, 1994.

[4] `http://scikit-learn.org/stable/`. (Last visited 03/20/18).

[5] `http://archive.ics.uci.edu/ml/datasets/volcanoes+on+venus+-+jartool+experiment`. (Last visited 03/20/18).