A universal model for the detection of stomata in plants

Van de Velde Arthur
Supervisor: prof. dr. ir. Francis wyffels
Counsellor: ir. Olivier Pieters
Ugent

ABSTRACT

***Abstract*—The future is in neural networks. The progress in computer power of the last few years made the way for great progress in neural networks. Because of this progress, neural networks are now able to recognize objects in an image. This is very useful for all kinds of applications.**

Biologists do research on stomata in plants. Stomata contain a lot of information. For example, it is possible for biologists to monitor the CO_2 concentration in the atmosphere using stomata. To perform these studies it is necessary to detect stomata on leaves. This is currently a very expensive and time-consuming process.

This paper investigates how the detection of stomata can be done automatically. To obtain this, a Faster R-CNN network is used. This is a neural network that was developed for object detection. The data needed to train the network will be examined. After this the network will be optimized and the results will be investigated.

***Index Terms*—Faster R-CNN, Neural network, Deep learning, Stomata, Object detection**

I. INTRODUCTION

Plant organisms are capable of building up their own carbon compounds. Plants will extract carbon dioxide (CO_2) from the atmosphere to build up these carbon compounds. The removal of CO_2 from the atmosphere is done through stomata. Stomata are present on the leaves of a plant. The stomata are able to open and close, when they are open CO_2 is extracted from the atmosphere.

The formation of stomata on the leaves of plants will depend on both short and long term conditions. For example, a leaf formed during a cold and wet winter will be large and will contain a lot of stomata. This while leaves formed in a dry and warm summer are rather small and contain little stomata [1]. The formation of stomata will also depend on long-term environmental factors. For example, plants that contain leaves with a lot of stomata will have a greater chance of survival when the CO_2 concentration in the air drops. Conversely, plants with few stomata will have a greater chance of survival when the CO_2 concentration in the atmosphere rises [1].

On the basis of these long-term effects it is possible for biologists to monitor the CO_2 concentration in the atmosphere and monitor its change. For example, the 400 million years during the Phanerozoic era, associated with a low CO_2 concentration, are characterized by leaves with a high stomata concentration.

Detecting stomata on leaves is therefore not unimportant for researchers. Currently, this is done manually, which makes it an expensive and time-consuming undertaking. However, thanks to recent advances in neural networks, it is possible to automate this process. The aim of the master thesis is to develop a model that is able to detect stomata on leaves. This model is not plant-specific but should work well on different plant species. To achieve this goal a Faster Region-based Convolutional Neural Network (Faster R-CNN) is used.

In part 2 the data obtained and how it is processed to train and test the neural network will be discussed in more detail. Part 3 explains how the Faster R-CNN network works. Part 4 examines the metrics. In part 5 the parameters are listed with their optimal value. In part 6 the results are reviewed. In part 7 the implementation is briefly mentioned and in part 8 everything is summarized.

II. DATA EXPLORATION AND PREPROCESSING

A first and important step in any neural network is the data. Without data it is not possible to train a neural network and without the correct data this training will go nowhere. In this section we will go deeper into the data used and how this data has been processed to train the neural network correctly.

A. Used data

The neural network aims to detect stomata on images of plants. To train the network for this, images of stomata on different plants are needed. An example of such an image is shown in figure 1. The complete data set consists of 113 different plant species combined with a .txt file on which the locations of all stomata are given. The .txt file contains the name of the plant species, combined with the x and y coordinates of the stomata and the size of the stomata.

The original data consists of RGB images of 1600 by 1200 pixels. These images were taken manually with a microscope. It is possible that there are differences between the images. For example, an image may be overexposed or underexposed. It is also possible that some images are less sharp or even blurred. Both fresh and dried materials are used. This can result in a difference in color. The fresh material will be greener than the dried material. Some plant species have an orange glow which differs from the green of most other plants. To deal with all these differences data preprocessing has been used.

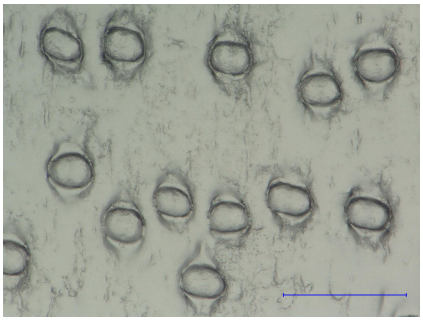

Figure 1: Example image of the data.

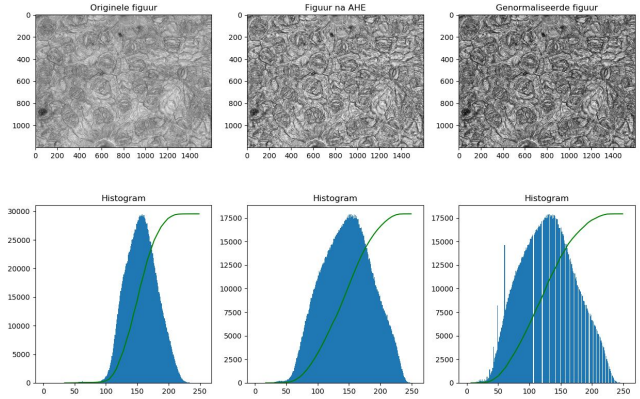

Figure 2: The images with their histogram at the different steps of the normalization process.

B. Generating data

An original image is shown in figure 1. This is an RGB color image of 1600 by 1200 pixels. For the neural network, images of 800 by 600 pixels are used. When generating data, it is therefore important to crop each image to 800 by 600 pixels. This can be done by cutting the image into four pieces.

It is also important that the model receives an equal amount of data for each plant species. Therefore, 200 stomata are generated for each plant. The original data includes plants for which sufficient images are available. With these plants it is therefore not necessary to generate extra data. Other plants only have a limited amount of data available. These plant species don't have 200 unique stomata. Here overfitting is used, this means that extra information is generated based on the original data. This happens in different ways. In a first phase the original image is mirrored. First over the x-axis, then over the y-axis and then over both axes. Then the image will be rotated 90 degrees and all reflections will repeat themselves. In this way it is possible to go from one image to 8 images. If there are less than 200 stomata, the image will be rotated. The image is rotated in an infinite loop with 25 extra degrees each time. With each rotation a piece of 800 by 600 pixels will be cut out of the image. This until 200 stomata are found.

C. Data preprocessing

Data preprocessing will ensure that the differences between the images are eliminated as much as possible. Data preprocessing is used on the training and testing data. However, the data produced for training purposes will differ from the data used for testing. For network testing the data will be normalized. The goal of normalizing data is to make all data as equal as possible. For example, it will ensure that images that are overexposed or underexposed will be adjusted so that the exposure becomes more central. When training a network, data augmentation is done. Here the normalized data will be slightly adjusted to get a robust data set. These minor adjustments ensure that a network can handle a wider spectrum of data.

D. Data normalization

The goal of data normalization is to get each image as consistent as possible. This makes it easier for the neural network to detect stomata. The normalization of data is done with three techniques. A first and simple technique is to convert the RGB color images to grayscale images. This conversion ensures that the differences in color between fresh and dry material as well as the differences between the green and more orange plants are largely reduced.

A second technique uses *Adaptive Histogram Equaliztion (AHE)*. The histogram is a graphical representation of the intensity distribution of an image [2]. With a grayscale image, the histogram will scan all pixels, look at their intensity, and store them cumulatively by their intensity value. AHE is used to improve contrast within an image. This is done by spreading the histogram of an image. The histogram before and after AHE conversion can be seen on the left and middle image in figure 2.

The latter technique uses the *gamma transformation*. The goal of this transformation is to get the average pixel intensity at 128. This will center the exposure of an image. The gamma transformation therefore mainly affects overexposed and underexposed images. The final image after normalization can be seen in the right part of figure 2.

E. Data augmentation

Data normalisation is used for the test data and aims to get this data as constant as possible. Data augmentation is used on the training data and will make adjustments to the normalized data, making the model more robust.

A first technique will narrow or broaden the histogram. A factor alpha is used to multiply the intensity of each pixel. The factor is randomly chosen between 0.85 and 1.15. After multiplication, a gamma transformation is used to get the average intensity back to 128.

Secondly the image will go through a gamma transformation. A random gamma is choosen between 0.75 and 1.25. A final technique will stretch the image along the x or y axis. A random number is chosen between 1 and 1.2. Next, the x or y axis is stretched by the length of the random number.

III. Faster R-CNN

A Faster R-CNN is chosen as the basic structure for the neural network. Different structures exist but the Faster R-CNN network is a network that is fast and yet is not just

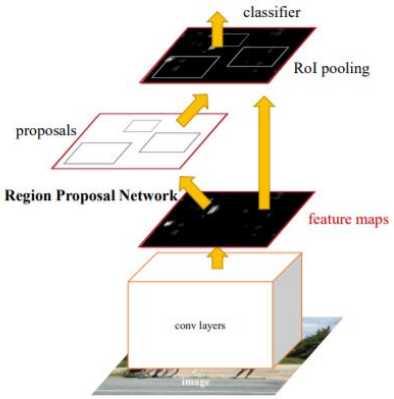

Figure 3: Brief structure of the Faster R-CNN network [4].

designed for the speed [3]. Faster R-CNN is a continuation of the R-CNN network that was developed to recognize objects on images with high accuracy [4]. As the name suggests, Faster R-CNN will be a faster version of the R-CNN network.

The Faster R-CNN network (figure 3) consists of three separate models that work together. The first part are 'the shared layers', these layers are shared by the next two models. A second model is the Region Proposal Network (RPN). This network transfers the interesting regions to the third model. That third model is responsible for the classification and is called the classifier.

A. *Object classification networks*

The operation of an object detection network is best explained by using figure 4 and figure 5. Figure 4 shows a VGG16 network. This network works in the same way as the ResNet-50 network but with slightly modified layers. The network is used to classify images. As shown in the figure, an image will enter the network. This image will pass through several layers and after each layer a new feature map will be created. The dimensions of these feature maps become smaller and smaller each time until at the end a 1x1x1000 map remains. So the entire image is shrunk to a 1x1x1000 matrix. Where the 1000 stands for the number of different classes the image can belong to. And the 1x1 stands for the one digit that remains. This is a digit between 0 and 1 that indicates the probability that an image belongs to the given class. This VGG16 network is used to classify an image between 1000 different classes.

The network in figure 4 consists of several layers. Each of these layers is important, but to understand how the feature map can recognize patterns and objects, the convolutional layer is especially important. This layer consists of a series of filters who will slide over the incoming pixels. In the first layers the filters are able to detect straight lines. As more filters are placed one after the other, the filters will be able to detect more complex objects. Figure 5b shows two feature maps of the fifth convolutional layer. The bottom feature map will activate very strongly to angles on an image. The arrow indicates which part of the feature map is activated the most.

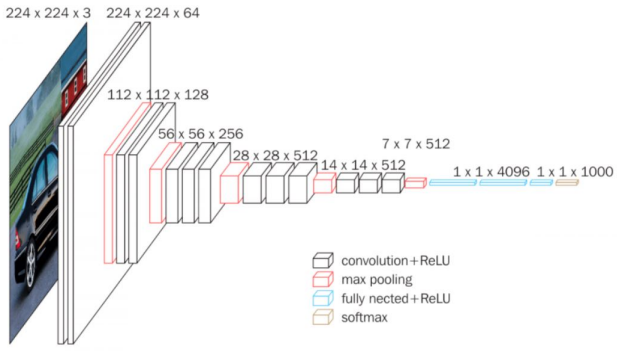

Figure 4: Schematic representation of a VGG16 network [5].

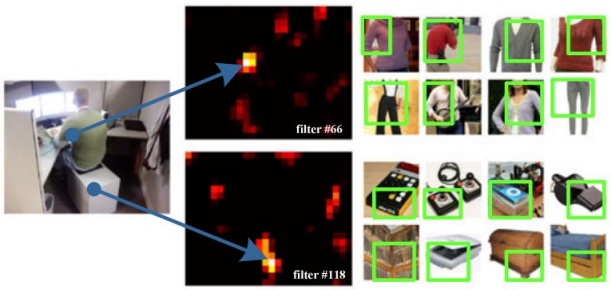

Figure 5: Presentation of feature maps. (a) An example image (b) Some feature maps of the conv $_5$ filters (c) Some other images that respond strongly to these filters. [6].

The operation of object classification networks should now be clear. In the next paragraph the Faster R-CNN network will be examined more closely. This is an object detection network which makes it slightly different from the object classification network.

B. *Object detection network*

The object detection network must do the same as the object classification network but on different parts of the figure. With Faster R-CNN these different parts are taken structurally using 9 anchor boxes (figure 6). These anchor boxes are taken at different points in the figure. A stride of 16 is used. This means that the 9 anchor boxes are taken every 16 pixels to insert into the network. All points where the anchor boxes are taken are shown in figure 7. On a 600x800 pixel image with a stride of 16 where 9 anchor boxes are taken at a time, the network will

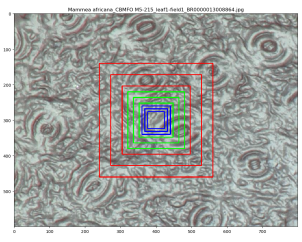

Figure 6: Square anchors.

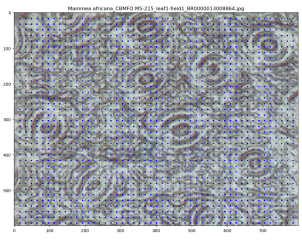

Figure 7: Grid for anchor boxes.

Gewenst	True positive	TP	Detects stomata correctly
	True negative	TN	Detects background correctly
Ongewenst	False positive	FP	Detects background as stomata
	False negative	FN	Forget to detect a stomata

Table I: Summary of the building blocks for testing a neural network.

generate 16200 anchor boxes. All the anchor boxes go into the shared layer and the feature maps are generated for each anchor box.

The shared layers are used by both the RPN network and the classifier. Because the two networks share the layers, it will not be needed to generate the feature maps twice. Therefore the network will be faster. The shared layers first create feature maps for each of the 16200 anchor boxes. These feature maps all enter the RPN network. The purpose of the RPN network is to distinguish the foreground from the background. This network will not look in which class an box belongs. Of the 16200 anchor boxes the RPN network will forward the best 80 to the classifier. The classifier is now trained to divide the incoming boxes into different classes. In this network the classifier will choose between 'stomata' or 'no stomata'.

IV. Metrics

The Faster R-CNN network consists of three parts, each with its own parameters. In order to set these parameters correctly it is necessary to test the network several times and compare these tests with each other. The comparison of the different tests is mainly based on the F1-score. The F1-score is a combination of the recall and the precision. The parameters rest on four building blocks which are therefore discussed first.

A. Building blocks

The four building blocks are described in table I and are shown in figure 8. The figure contains text for the different boxes. The text indicates the type of building blocks the box represents and the prediction of the model on the box. Also note that there is text for a stomata that is not surrounded by a box. This is because the model did not find this stomata.

B. Precision

Precision is calculated using the following formula.

$$Precision = \frac{TP}{TP + FP}$$

Precision is the number of correctly detected stomata divided by the number of boxes that the network thinks contain a stomata. So how correct the frames are that are indicated as stomata. The problem with precision is that the network on a 20 stomata image can have a precision of 100% when the model predicts one stomata and nothing else. The model does not make any mistakes in this case but will still miss 19 stomata. The precision doesn't takes the False Negatives into account.

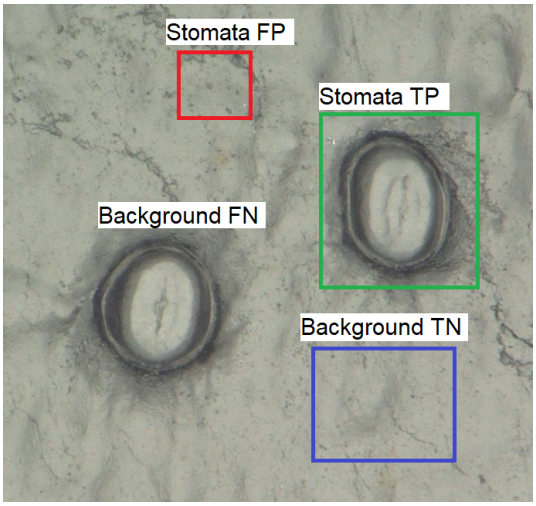

Figure 8: Visualization of the building blocks.

C. Recall

Recall is calculated using the following formula.

$$Recall = \frac{TP}{TP + FN}$$

Recall is the number of indicated stomata divided by the total number of stomata present. The recall score will therefore indicate how good the network is in finding all the stomata. The problem with recall is that the network on an image with 20 stomata can have a 100% recall when this model indicates 300 frames as stomata and so happens to indicate the 20 real stomata.The recall does not take the False Positives into account.

D. F1 score

To address the problems of precision and recall, the F1 score is used [7]. The F1 score is calculated using the following formula.

$$F1 = 2 * \frac{precision * recall}{precision + recall}$$

By combining precision and recall, the network will be penalized if it indicates too many stomata incorrectly or when it finds too few stomata. Because of this, the F1 score is often used in the validation of neural networks.

V. Exploration of hyperparameters

The Faster R-CNN network consists of several parameters. It is also possible to train on different types of data. During the exploration of the hyperparameters, the optimal parameters are searched to train the network. This is done by adjusting one parameter each time and comparing the results with the previous network. In this part, the optimal hyperparameters are discussed. The hyperparameters have not yet been explained. In order to understand them, background knowledge about Faster R-CNN networks is expected.

After the many tests it appears that the Faster R-CNN network will train optimally on slightly augmented data. The

F1	Bij optimale threshold		Bij algemene threshold (0,966)	
[0 ; 0,5[	4		16	
[0,5 ; 0,8[	33		38	
[0,8 ; 0,85[	13	76	21	59
[0,85 ; 0,9[	19		9	
[0,9 ; 0,95[	21		19	
[0,95 ; 1]	23		10	

Table II: General results of the F1 scores after testing the general model.

techniques and parameters used for light data augmentation are explained in section II-E. The complete data set of 113 plant species contains 10723 images. These images are divided into a part for training and a part for validation. This is done according to a 90-10 split where 90% of the data is used to train the model and 10% is used to validate the model. As test data, 2500 random images are selected from the complete data set. This is from a data set of normalized images. So testing is done on normalized data.

The training of the model is done in two parts. First, the network will train both the RPN model and the classifier. This part will train for 10 epochs. Next the model only trains the classifier for another 10 epochs. At the start of the training an already trained network is loaded in. Because of this it is no longer necessary to train all layers. The first 80 layers are not trained and keep their initial values. The Faster R-CNN network has 143 shared layers. So 80 layers corresponds to 56% of the shared layers.

The learning rate is set to 0.0001. The RPN model will be able to pass a maximum of 80 boxes to the classifier. And the classifier batch size is set to 16. The weight of the unloading functions is set to one.

VI. Results

After the optimal hyperparameters have been examined, the results of the network will be analysed. The examination of the results is divided into a qualitative study and a quantitative study.

A. Quantitative analysis

The F1 scores of the different plant species are plotted in table II. It is assumed that the network can detect stomata correctly when the F1 score is above 0.8. The left part of the table shows the scores when the threshold for each plant species is set separately. There are 76 plant species with a good F1 score. There are also 33 plant species of which the F1 score is not good. But for these 33 plant species the model can still be optimized. There are also four plant species with an F1 score below 0.5. These four plant species have stomata that cannot be detected with a general model. It is possible to train a model specifically for these plant species. But the goal here is to make a global stomata detector.

Within the quantitative research it is possible to investigated whether the size of the stomata has influence the correctness of the detection. In addition, it is also examined whether the oversampling of data causes problems.

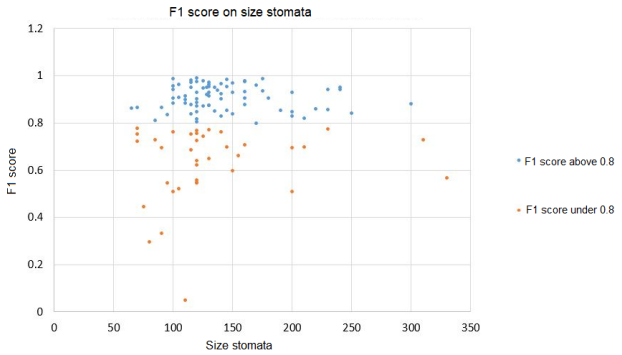

Figure 9: Distribution graph of the F1 score in relation to the size of a stomata.

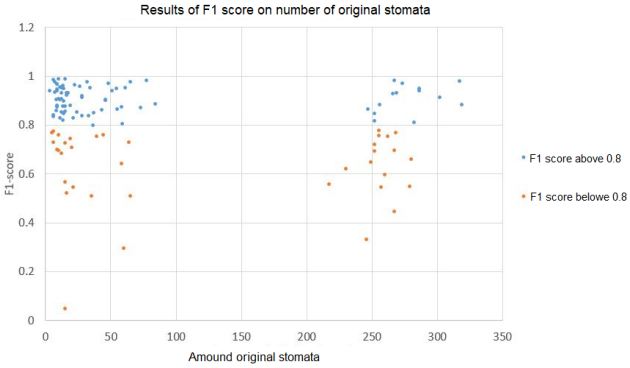

Figure 10: Distribution chart of the F1 score compared to the original number of stomata.

1) Size of stomata: In figure 9 the F1 scores are plotted relative to the size of a stomata. This shows that there is no correlation between the size of a stomata and the F1-score.

2) Oversampelen: Not all plant species have 200 stomata in the original dates. Some species contain more than 200 stomata and others contain only a few stomata. It is possible that the preprocessing technique used, for generating more stomata, does not provide sufficient variation in the data. In figure 10 the number of original stomata are plotted on the F1-score.

From figure 10 can be concluded that the assumption is wrong. It is not the plant species with little original data that are doing badly, but those with a lot of original data. Of the 33 plant species, 15 have more than 200 original stomata. This is 45,45%, while only 14 of the 76 plant species have more than 200 stomata. This corresponds to 18.42%.

B. Qualitative analysis

Qualitative analysis shows that stomata that stand out against their background are better detected. A background that contains little information helps the stomata to stand out from the background (figure 12). Therefore, it is more difficult to detect stomata with a busy background (figure 11).

C. Oversampelen

The network currently works well on 76 of the 113 plant species. The aim of the thesis is to develop a network that

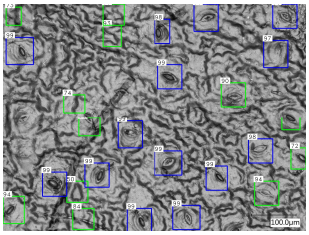

Figure 11: Dialium pachypyllum, plant species with busy background.

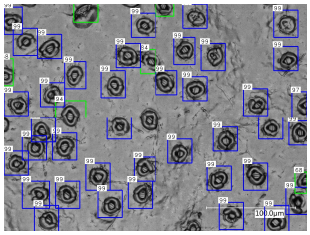

Figure 12: Paramacrolobium coeruleum, plant species with flat background.

F1	Optimal threshold		Global threshold (0,969)	
[0 ; 0,5[	6		11	
[0,5 ; 0,8[	26		35	
[0,8 ; 0,85[	16	81	22	67
[0,85 ; 0,9[	27		15	
[0,9 ; 0,95[	22		22	
[0,95 ; 1[	16		8	

Table III: General results of the F1 score with extra training on the 33 plant species.

works well on as many plant species as possible. There are different ways to train extra on a part of the data set. Undersampling, oversampling thresholding or cost sensitive learning can be used.

The choice was made to oversample with a factor of two. This means that the 33 plant species will occur twice as much in the training data set. After training with oversampling, the results from table III are obtained. However, the table alone gives a wrong picture. Of the original 33 plant species, 15 now have an F1 score above 0.8. And the F1 score will increase on 25 of the 33 plant species. This with an average of 0.129. However, 10 new plant species have been added that previously had an F1 score greater than 0.8. This can be explained, because of the extra testing on the 33 plant species there is less testing on the plant species with previously a good F1-score. So the F1-score of these plants will decrease slightly.

In general, this new model is better. In a further phase it would be possible to oversample the 10 species of which the F1-score dropped below 0.8. This oversampling will not happen with a factor of two, but rather with a factor of 1.5. It can be further investigated whether this works.

VII. Implementation

The aim of the thesis is to help biologists detect stomata. Biologists should therefore be able to use the developed neural network. To achieve this, a site will be developed and the network will be placed on a server of Ghent University. To secure the servers, the site is placed in a docker container.

The intention is to analyze multiple images at the same time. With this idea in mind the choice has been made to work with zip files. The images that have to be analysed will be put in a .zip file. The zip file is then uploaded to the servers. Here the neural network will analyze the images and put boxes around the stomata. These are then zipped together with a .txt file and sent back to the user. The .txt file contains all the names of the images that were uploaded combined with their stomata.

VIII. Conclusion

In this thesis several neural networks have been investigated that could help biologists detect stomata. Here the choice was made to work with a Faster R-CNN network for the detection. And with a ResNet-50 network as backbone. In order to optimise the use of this network for stomata research, different parameters and metrics were used. These were extensively tested in order to arrive at an ideal combination. The after this the network was tested on the full data set. Here a qualitative and quantitative investigation was used to determine why some stomata were correctly detected and why it was more difficult for the network to detect other stomata. After this investigation one more technique was used to optimise the network.

On the final network 71% of the plants have a F1 score above 0.8. Which means in 71% of the plants the stomata will be detected correctly, or at least good enough to be useful. With further optimization it should be possible to get this number even higher. The final network is then put on the servers. This makes it possible to use this network from all over the world.

References

[1] A. M. Hetherington and F. I. Woodward, "The role of stomata in sensing and driving environmental change," *Nature*, vol. 424, no. 6951, pp. 901–908, 2003.

[2] M. Shin, M. Kim, and D.-S. Kwon, "Baseline cnn structure analysis for facial expression recognition," in *2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pp. 724–729, IEEE, 2016.

[3] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, *et al.*, "Speed/accuracy trade-offs for modern convolutional object detectors," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7310–7311, 2017.

[4] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, pp. 91–99, 2015.

[5] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.

[7] T. Saito and M. Rehmsmeier, "The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets," *PloS one*, vol. 10, no. 3, 2015.