UNIVERSITY OF TARTU
Faculty of Science and Technology
Institute of Technology

Ivan Gorbachenko

Automated Detection and Quantification of Stomata

Bachelor's thesis (12 ECTS)

Supervisor:

Hanna Hõrak

Co-supervisor:

Rain Eric Haamer

Tartu 2024

Resumee/Abstract¨

Automated Detection and Quantification of Stomata

Abstract:

This thesis presents an approach for the automated detection and quantification of stomata using machine learning techniques. The study focuses on employing the YOLOv8 model to analyse video data of leaf epidermal imprints, significantly improving the efficiency and accuracy of stomatal detection compared to traditional manual methods. The results highlight the model's ability to handle varying focal depths within video frames, ensuring consistent stomatal counts. Future research directions include expanding the dataset and incorporating advanced image analysis techniques to further enhance detection accuracy.

Keywords: Stomatal detection, machine learning, YOLOv8, plant phenotyping, image analysis

CERCS: Machine learning

Õhulõhede automaatne tuvastamine ja loendamine

Käesolev lõputöö loob lähenemise taime õhulõhede automaatseks tuvastamiseks ja loendamiseks masinõppe tehnikate abil. Uuring keskendub YOLOv8 mudeli rakendamisele lehe epidermi jäljendite videote analüüsimisel, mis oluliselt parandab õhulõhede tuvastamise efektiivsust ja täpsust võrreldes traditsioonilise käsitsi loendamisega. Tulemused näitavad, et mudel suudab analüüsida õhulõhesid videokaadritel), tagades järjepideva õhulõhede loendamise kogu videos uuritavalt alalt, kus erinevad õhulõhed on fookuses erinevatel videokaadritel. Tulevased uurimissuunad hõlmavad andmekogumi laiendamist ja täiustatud pildianalüüsitehnikate integreerimist, et veelgi parandada õhulõhede tuvastamise täpsust.

Võtmesõnad: Õhulõhede tuvastamine, masinõpe, YOLOv8, taimede fenotüpiseerimine, pildianalüüs

CERCS: Masinõpe

Contents

Terms, abbreviations, and notations

CO_2 - Carbon dioxide

ML - Machine Learning

DL - Deep Learning

ANNs - Artificial Neural Networks

CNNs - Convolutional Neural Networks

DCNN - Deep Convolutional Neural Network

YOLOv8 model - You Only Look Once version 8 model

FCN - Fully Convolutional Network

Mask R-CNN - Mask Region-Based Convolutional Neural Network

AP - Average precision

NDVI - Normalised Difference Vegetation Index

SGD - Stochastic Gradient Descent

IQR - Interquartile Range Filter

VDP - Versatile Data Pipeline

2D FFT - Two-dimensional Fast Fourier Transform

Introduction

At the University of Tartu's Molecular Plant Physiology Lab, one of the responsibilities assigned to researchers involves analyzing the epidermal imprints of leaves. Their work, which is essential for understanding how frequent respective cell populations are in different plants, depends on the quantification of stomata.Stomata are small pores in the leaves, distinguished by their elliptical shape and a noticeable division in the center, resembling an eye-slit.

Leaf epidermal impressions collected in the lab are usually not flat enough to allow the quantification of stomata from a single image obtained under the microscope. Therefore, researchers record a video while gradually adjusting the focus through various depths of the sample. This method ensures that all stomata come into clear view as the focal plane moves up and down. In the lab's current method, stomata are manually quantified from videos using ImageJ software (Figure 1) [1]. This process, however, is quite time-consuming. It involves opening the video file in ImageJ software as a stack of images, navigating between different focal planes to identify stomata, and then manually labeling each stoma.

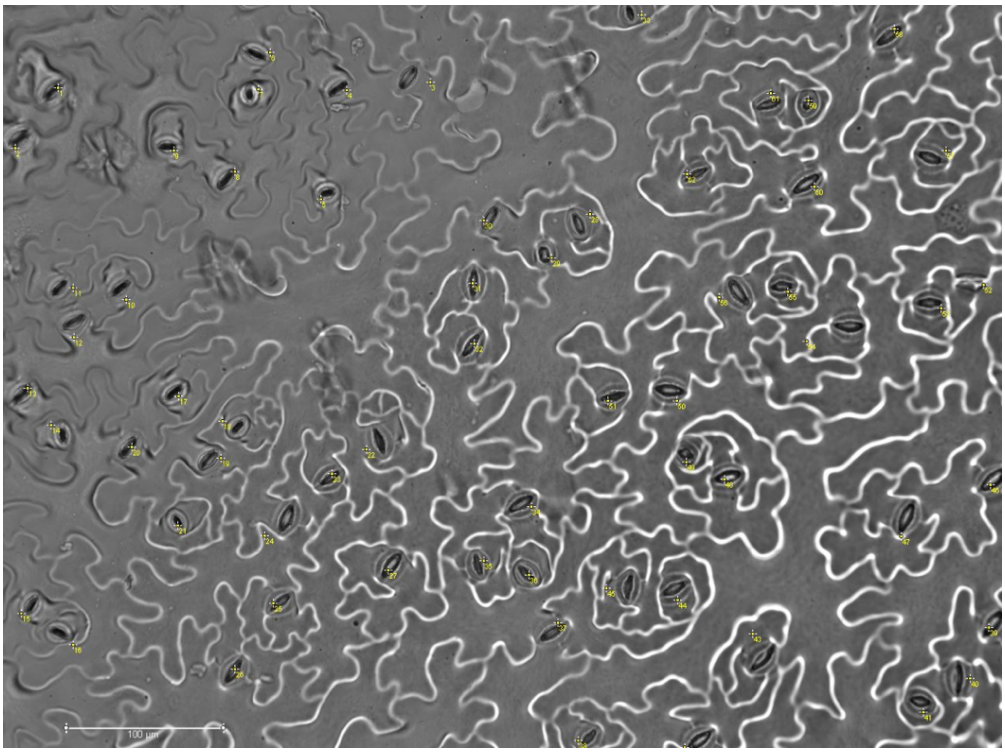

Figure 1: Image with labeled stomata in ImageJ.

Developing a machine learning approach to automatically detect and quantify stomata from video files could significantly improve and speed up the current process for leaf impression analysis. The aim of my thesis is to create an algoritm that will analyze video files with stomata and provide an accurate stomatal count. By impleenting such approach, quantitative analysis in the lab can become more consistent, as different people may achieve somewhat different results by doing it manually.

1 Background

This chapter introduces the key concepts required to understand the thesis. It provides an overview of stomata, bright-field microscopy for stomatal imaging, digital image processing, image analysis for plant phenotyping, machine learning and deep learning

1.1 Stomata

Stomata are small openings found on the outer layer of plant aerial structures, such as leaves and stems. They derive their name from the Greek term meaning mouth. Each stomatal pore in plants is surrounded by a pair of subsidiary cells and a pair of guard cells that typically resemble the shape of kidneys, however, in monocot grasses like barley, they have a dumbbell form (Figure 2) [2].

Stomatal pores are crucial for plant physiology as they regulate the absorption of carbon dioxide (CO_2) into leaves, which directly affects the rate of photosynthesis. In addition, stomata mediate transpiration and regulate plant water loss, hence controlling leaf cooling and water use efficiency, which is vital for plants to survive drought stress.

Given the significance of stomata, studying stomatal control has become an increasingly common task for biologists [3].

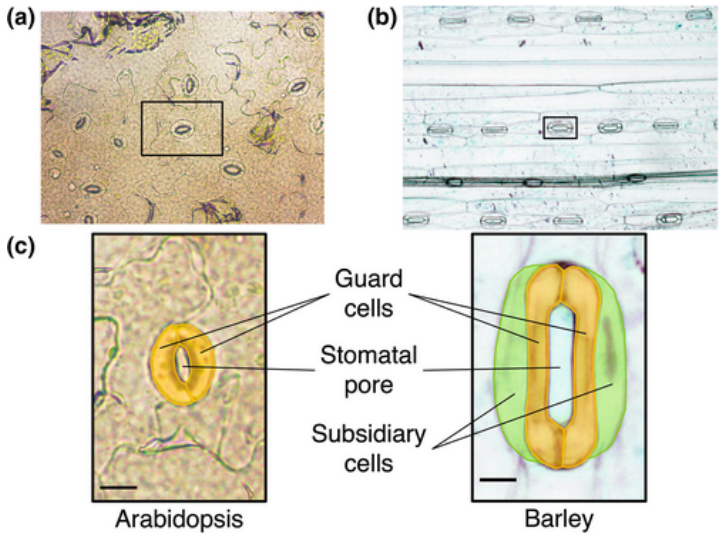

Figure 2: Representative captured Arabidopsis (Arabidopsis thaliana) (a) and barley (Hordeum vulgare) epidermis. Stomata of Arabidopsis thaliana and barley (c) [2].

1.2 Bright-field microscopy for stomatal imaging

Bright-field microscopy is an essential visual technique that illuminates the transperent nail varnish leaf imprint to get a distinct and well-defined image of a tiny structures within it, including leaf epidermal cells [3].

The preparation for bright-field microscopy starts with creating silicon impressions from the leaves, then nail polish is applied to create imprints from the silicon molds. Varnish impressions are transferred to microscope slides using transparent tape, creating samples suitable for bright-field microscopic imaging [4].

This method allows for the efective and quick acquisition of videos of epidermal leaf impressions with visible stomata on them (single frame from a video shown in Figure 3).

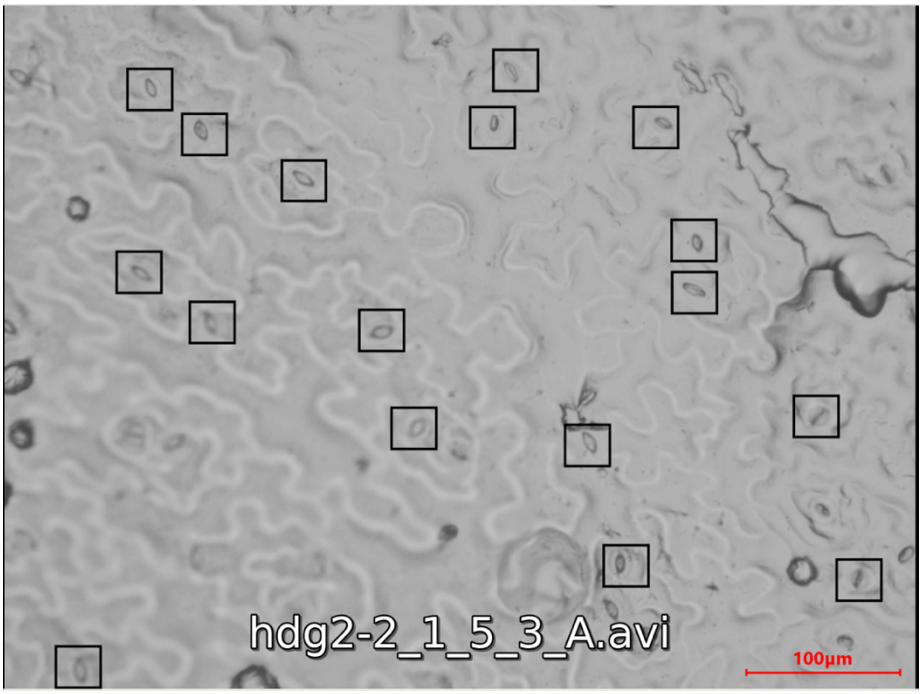

Figure 3: Single frame from the video sample with stomata

1.3 Digital image processing

A digital image is classified based on the number and type of values each pixel contains. There are three different types of images: binary, grayscale, and color image. A binary image has only two possible values for each pixel, typically 0 or 1, representing two distinct colors. Commonly, these colors are black and white, but any two colors can be used. A grayscale image, in contrast, carries intensity information for each pixel, representing various shades of gray that range from black (the lowest intensity) to white (the highest intensity). Lastly, a color image requires three values per pixel, corresponding to the red, green, and blue (RGB) color channels, to represent a wide range of colors [5].

Digital image processingis used to perform operations on the image with the purpose of enhancing it or extracting valuable information. This methodology is commonly used in various fields of science and technology, including biological research [6].

Digital image processing entails a variety of operations that can manipulate images to improve visual quality and detect specific features. In the context of my thesis, several key image processing techniques were used: convolution, two-dimensional Fast Fourier transform (2d FFT), band-pass filtering, and thresholding.

1.3.1 Convolution

Digital image processing uses convolution as a basic method. Convolution is the process of changing a picture by applying a kernel, commonly referred to as a filter, to the original image. A kernel is a matrix that moves over the image, multiplying the overlapping pixels element-by-element and adding the results to create a new pixel value in the final image [7].

Different kernels are made for different uses. For example, blurring kernels are employed to minimize noise and smooth out images. A Gaussian kernel assigns weights based on the Gaussian function, giving higher importance to the central pixels and less importance to those further away, resulting in a blurred effect. Edge detection kernels are used to highlight edges within an image. Edge detection kernels highlight areas with significant variations in intensity.

1.3.2 Band-pass filter

A band-pass filter is essentially a combination of a low-pass filter and a high-pass filter. This type of filter allows frequencies within a certain range to pass through while blocking frequencies outside of this range, leaving only objects of interest visible. A high-pass filter allows high frequencies to pass, which helps in sharpening the image by emphasizing edges and fine details. A low-pass filter allows only low frequencies to pass, which helps reduce noise and smooth out variations in the image.

1.3.3 Two-dimensional fast fourier transform (2d FFT)

The 2d FFT is a powerful tool used in image processing for analyzing the frequency components within an image. When we apply a 2d FFT to an image, we transform this image from the spatial domain into the frequency domain. An image in the spatial domain is a matrix where each pixel has a value representing some property, like color or brightness. In a grayscale image, for example, the value of a pixel is just the intensity of black or white, where black is represented by value 0 and white by value 255. The frequency domain represents the image in terms of the rates at which pixel intensities change across the image. High frequencies in the image correspond to rapid changes in pixel intensity, which often appear at the edges of objects. Low frequencies represent smooth transitions in intensity and cover areas with less change in color and intensity. This transformation can reveal shapes in the image that are not visible in the regular spatial view, particularly periodic structures like stomata, which may have a repetitive spatial frequency.

1.3.4 Thresholding

Thresholding is a technique for image segmentation that involves transforming a grayscale image into a binary image by selecting a threshold value. Pixels with intensity values above this threshold are set to one (representing white), and those below are set to zero (black), thus distinguishing the foreground from the background. This method is widely used because of its simplicity and effectiveness in scenarios where the object of interest contrasts well with the background.

1.4 Image analysis for plant phenotyping

The scientific community has developed numerous image analysis tools specifically designed for high-throughput plant phenotyping using image-based techniques [6], [8] and [9] . These technologies vary in plant characteristics that they they can measure.

Standard image processing pipelines have provided acceptable results for measuring phenotypic features. As an example, estimation of biomass [10], evaluation of normalised difference vegetation index (NDVI) [11] and chlorophyll responses [12] can be achieved by segmenting a plant from a known background. Further, such phenotypic features as plant height, leaf area, and shape also can be measured by using straight-forward processing steps after plant segmentation [13].

However, complex phenotyping tasks, including plant stress response, disease symptoms, age estimation, yield prediction, stomatal counting and other high level tasks require more sophisticated image processing techniques with several more steps including morphological operations, connected components analysis, and others [14].

Existing image-based phenotyping tools that depend on image processing pipelines, require many filters and manual adjustment of parameters due to changes of brigthness, contrast, and exposure of the sample image, making this approach ineffective [15]. Machine learning techniques, and deep learning in particular, have the potential to succeed better in stomatal detection.

1.5 Machine learning (ML)

In response to the limited flexibility of classical image processing pipelines for complex phenotyping tasks, machine learning techniques are taking a prominent role in the image-based phenotyping [16].

Machine learning (ML) is a branch of artificial intelligence (AI) that creates algorithms that are able to learn from a prior data and improve learning over time by using more data. Machine learning algorithms can make predictions or decisions about new data without being specifically programmed to do it. It is done my mapping features from an inut data into predicted labels.

We can classify machine learning into two main categories: supervised learning and unsupervised learning. Supervised learning involves training the algorithm on labeled data to create a model that

predicts labels for new data. Unsupervised learning, on the other hand, uses unlabeled data to discover hidden patterns or features within the data, such as clustering [17].

1.6 Artificial Neural Networks (ANNs)

Artificial Neural Networks are a subclass of machine learning models inspired by biological nervous systems. It consists of connected processing units known as neurons, which have the ability to learn patterns from data. Every neuron receives input signals, processes them, and produces an output signal, just like biological neurons do. The input signals are usually represented as a vector of values, which are then processed using a weighted sum and a non-linear activation function [18]. Weights are the parameters that the network learns during the training process. During the training process, the network fine-tunes these weights in order to minimize any errors in its predictions.

Neural networks are structured into different layers. The input layer is responsible for receiving input data and passing it on to the next layer without any processing. Hidden layers come next and process inputs from the previous layer using weighted connections and activation functions. Finally, the output layer generates the network's final output, with the number of neurons matching the number of prediction classes or regression outputs.

The process of training artificial neural networks (ANNs) involves a learning process where input data is passed through the network layer by layer. During this process, weights and activation functions are applied to produce the final prediction. The prediction is then compared to the actual target values using a loss function to measure the error. During the process of backward propagation, the network fine-tunes its weights in order to minimize the loss. This is achieved by calculating the gradient of the loss function with respect to each weight and then adjusting the weights accordingly. This process, repeated over numerous iterations (epochs), continues until the network's performance stabilizes or achieves a desirable level [18].

1.7 Deep learning (DL)

Deep learning is a subset of machine learning that focuses on using neural networks with multiple layers to analyze complex data features and provide a deeper understanding of the data. With an increased number of trainable weights, the model can capture nonlinear relationships and extract a wider range of features. Deep learning allows models to take raw data and automatically learn the features needed for detection and classification [19].

1.8 Convolutional Neural Networks (CNNs)

Convolutional Neural Networks (CNNs) are a type of deep learning model specifically designed to process and analyze visual data by applying convolution operations with various kernels to extract hierarchical features from the input. These features are then used to understand and categorize the data. CNNs are particularly well-suited for image-related tasks, such as recognizing features within images, including stomata [20].

1.9 YOLO model

The YOLO (You Only Look Once) model is a revolutionary approach in the field of object detection. It was created to offer real-time object detection capabilities while ensuring high precision. YOLO simplifies the object detection process by framing it as a single regression problem, going straight from image pixels to bounding box coordinates and class probabilities. This eliminates the need for complex pipelines and multiple stages of processing used in traditional methods.

One of the main advantages of YOLO is its capability to analyze the entire image at once. YOLO divides the input image into a grid and predicts bounding boxes and probabilities for each cell, allowing for the detection of multiple objects at the same time[21].

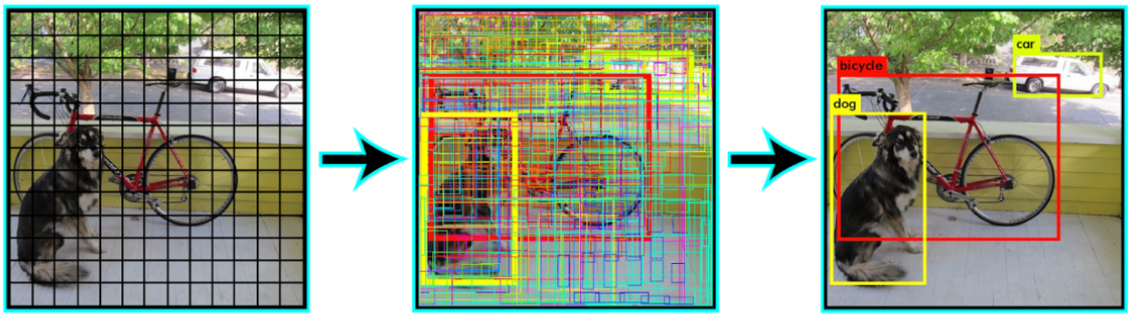

Figure 4: YOLO model working process [22]

YOLOv8 is the latest model that offers improved accuracy and efficiency over previous versions.The improvements in YOLOv8 make it particularly suitable for complex image recognition tasks [23]. For instance, it can be used effectively for detecting stomata in microscopic images, where precise localization and classification are crucial.

2 Related Works

To have a comprehensive understanding of the current stomatal detection solutions, it is necessary to examine the relevant published works. The following section presents four major machine learning and deep learning techniques that have been previously developed for the identification and quantification of stomata. An alternative new system is offered based on the examination of earlier works and advancements in technology after their publication

2.3 LeafNet

LeafNet is an advanced tool developed to automatically segment and quantify stomata and pavement cells in leaf images. This employs a hierarchical approach that applies deep learning to accurately analyze leaf epidermal images, especially those obtained under a bright-field microscope (Figure 4) [24].

LeafNet employs a two-step process involving StomaNet and LeafSeg modules. The StomaNet module implements a deep residual neural network consisting of three subnets, particularly in-scale, down-scale, and up-scale residual blocks that accurately identify stomata in input images. These subnets generate a probability heatmap of stomata presence and identify reliable stomata, which are then marked for the segmentation process. Once stomata are detected and masked, LeafSeg segments pavement cells using a region merging technique to properly outline the borders of pavement cells, even in the presence of noise and irregular shapes. As the result, we get an image with segmented stomata and pavement cells.

LeafNet's hierarchical strategy is illustrated in Figure 5. Figure 5A represents a normal bright-field image of the leaf epidermis, while Figure 5B illustrates the expected segmentation with stomata marked in blue and pavement cells highlighted in multiple colors. Figure 5C displays the expected statistical results, which include the size and count of stomata and pavement cells. Figure 5D depicts the training data that was created through manual segmentation, which is crucial for training the deep learning models. Figure 5E shows the sequence of steps in the workflow. Initially, the input image undergoes processing by the StomaNet module to identify stomata. Subsequently, the LeafSeg module is employed to segment pavement cells. Figure 5F presents a visual representation of the deep residual neural network employed in the StomaNet module.

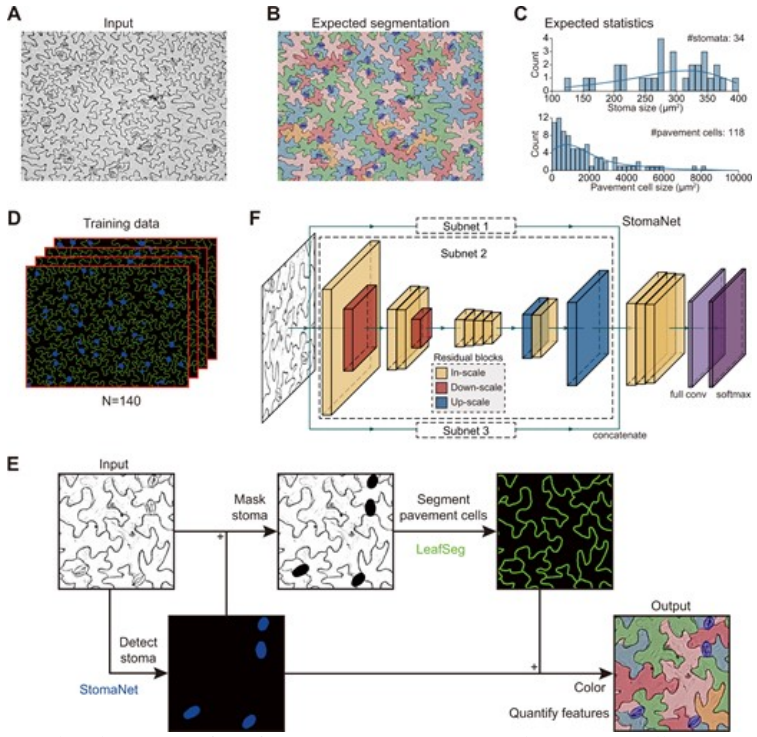

Figure 5: Hierarchical strategy of LeafNet to segment stomata and pavement cells [24].

Figure 6 provides a detailed graphical representation of the LeafNet workflow. Figure 6A shows a representative input image. Figure 6B demonstrates the StomaNet module generating a probability heatmap and detecting stomata, which are masked in the input image. Figure 6C illustrates the LeafSeg module's ability to identify potential edges and final segmentation output is shown in figure 6D.

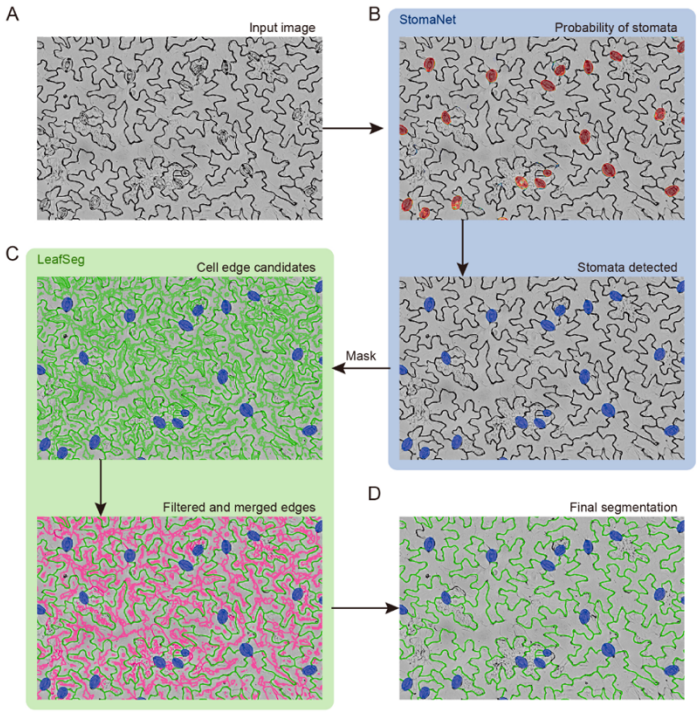

Figure 6: Graphical representation of the LeafNet workflow [24].

2.4 StomataCounter

StomataCounter is another tool designed to automate the identification and counting of stomata in microscopic images. It applies a deep convolutional neural network (DCNN) to achieve high accuracy in stomatal analysis. This tool allows plant biologists to upload plant epidermal image datasets to a pre-trained network and then identify and count stomata.

The workflow consists of several important steps (Figure 7). The process starts with the collection of human-labeled training images, where stomata are manually annotated and then used in the creation of a training dataset. Parts of images with non-stomatal regions are also included to train the network to distinguish between stomatal and non-stomatal areas. Further, these patches are extracted and augmented by rotating them into various orientations to increase the diversity of the training data.

Next, a deep convolutional neural network (DCNN) is trained on the prepared dataset. The network processes input images that are 224x224 pixels in size and generates a binary image output that indicates whether stomata are present or absent. The training process employs a stochastic gradient descent (SGD) solver to optimize the network's weights by considering the true labels of the training patches.

Once the initial training is complete, the weights acquired from the DCNN are transferred to a fully convolutional network (FCN). This adjustment enables the network to evaluate the probability of stomatal occurrence throughout the entire input image, rather than just in patches. As a result, it can effectively process images of any size. When processing the test images, FCN generates a prediction heatmap that shows the probability of stomata being present at each pixel location in the image. The prediction heatmap is then adjusted to eliminate low-probability detections, ensuring that only true stomata are considered. Stomata centers are identified by applying peak detection to

the thresholded heatmap. Excluding peaks within a predefined margin prevents edge effects and ensures accurate detection, even near the image borders. At last, the peaks that have been identified are added up, resulting in a precise and automated count of stomata in the input image. The identified stomata positions are projected onto the original image to enhance visualization and facilitate additional analysis [25].

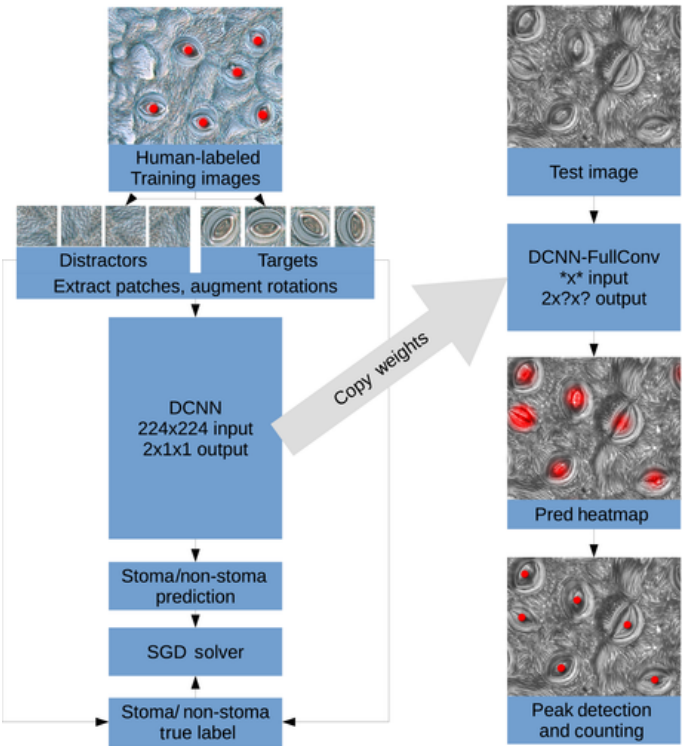

Figure 7: Architecture of the deep convolutional neural network (DCNN) and classification tasks [25].

2.5 StomaVision

StomaVion is one more recent tool designed for stomata detections and also measurment of its traits. This tool combines the YOLOv7 object detection model with a versatile data pipeline (VDP), creating a automated platform for plant scientists.

The workflow of StomaVision consists of several steps: generating instance masks, calculating rotated bounding boxes, deriving bounding box dimensions, detecting stomata, and integrating with VDP connectors (Figure 8) [26].

The YOLOv7-seg model generates example masks for the stomata and segments each stomatal pore. Every identified stomatal pore is enclosed within a rotated bounding box, allowing precise calculations of the aspect ratio and pore area. As a reasult, the output is annotated images that include bounding boxes and comprehensive statistical summaries including stomatal count, pore length, width, and area.

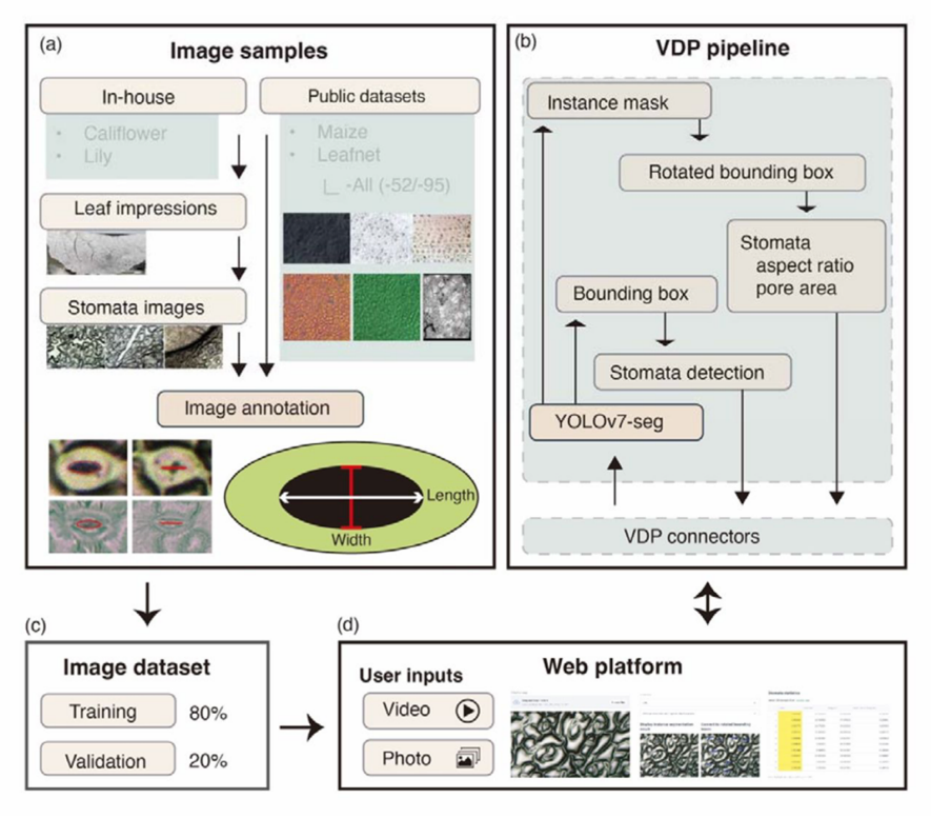

Figure 8: StomaVision training and inference workflow. (a) Image acquisition and stomatal annotation: Sample images were collected from independent sources to represent the different species. Graphical illustration indicating the manually labeled stomatal axes using red and white arrows and delineating the stomatal pore with a black mask. (b) Stomatal image inference: A versatile data pipeline (VDP) was utilized to streamline the integration of image processing and machine learning algorithms. (c) Annotated stoma images were allocated to training (80%) and validation (20%) datasets. (d) User Interface: A web-based platform capable of batch processing of various file types, including MP4 videos and PNG/JPG images. It presents a side-by-side view of the original and predicted images, along with a detailed output table enumerating stomatal parameters, such as ID, image dimensions, axis lengths, axis ratio, and pore area [26].

2.6 StomaAi

StomaAI is another innovative tool that accurately measures stomatal pores and density by using computer vision. The key innovation of StomaAI is its ability to accurately measure pores through a carefully designed inference and post-processing pipeline. The provided image illustrates the pipeline, which consists of multiple important stages (Figure 9).

The workflow starts by obtaining microscope images from the leaf surfaces of plants such as Arabidopsis and barley and, if necessary, converting them into supported formats like PNG, BMP, JPEG, and TIFF.

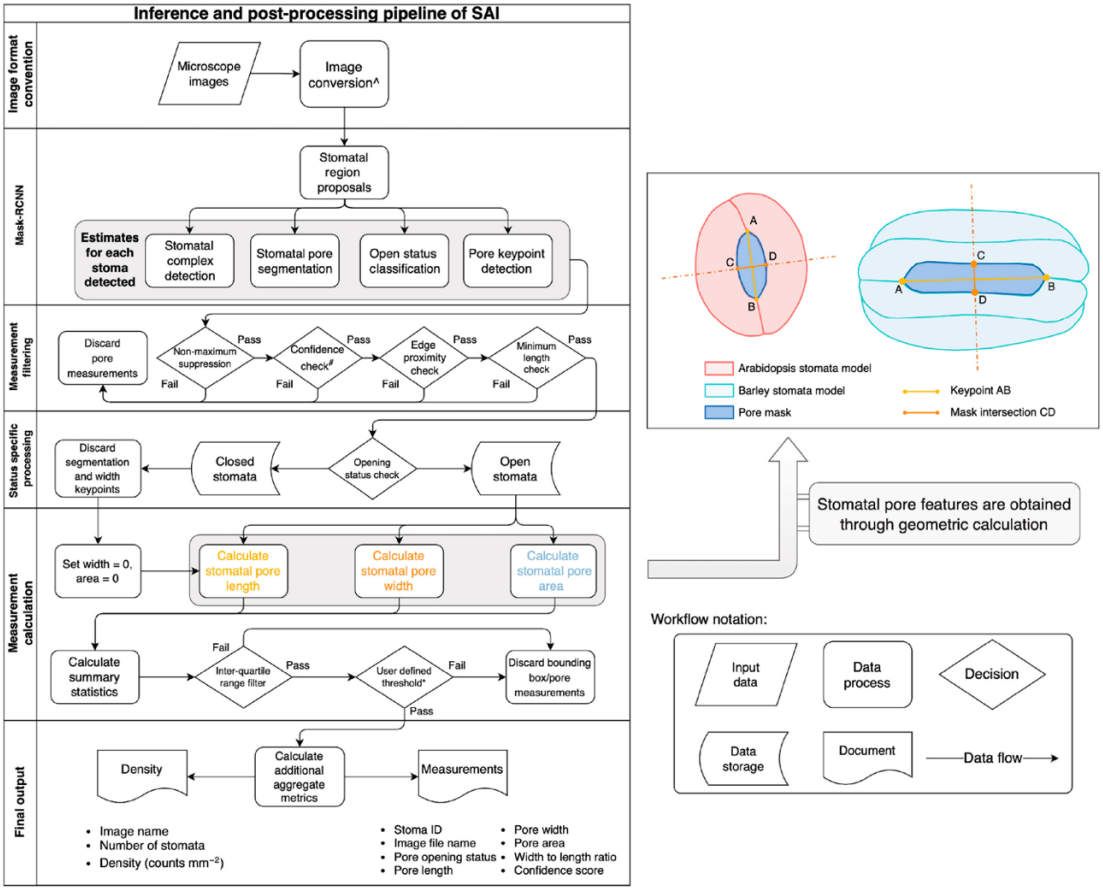

Figure 9: The inference and post-processing pipeline of StomaAI (SAI) [2]

StomaAI uses the Mask R-CNN (mask region-based convolutional neural network) framework to generate accurate proposals for pore measurements. The process starts with the region proposal network, which detects smaller image areas as regions of interest, each expected to have a single stoma. After generating these region proposals, they are then passed on to four distinct prediction heads, each dedicated to a specific type of measurement. The initial head generates a bounding box to further localize the stomatal complex, the second head determines the status of the stomatal pore, whether it is open or closed, third one attempts to mask the exposed area of the pore and the last one calculates the positions of two keypoints that indicate the start and end of the stomatal pore.

StomaAI also uses a series of post-processing steps to ensure the accuracy of measurements. It eliminates duplicate detections by employing non-maximum suppression, a technique that eliminates measurements with substantial overlap and the lowest confidence scores. Once the initial filtering is complete, additional measurements like the length of the pore can be estimated using keypoint detections. To determine the pore width, StomaAi finds points where the stomatal pore mask intersects with the mid-perpendicular line of the length keypoints.

After initial measurements, an interquartile range (IQR) filter is used to get rid of any outliers after the first readings. Samples whose pore lengths are significantly less than the median or whose bounding box measures are significantly greater than the median are not included. It is also possible to add more user-defined filters to change the confidence score level and leave out immature stomata based on how long they are.

The final output includes density and detailed measurements for each stoma [2].

Comparison of related works

All four tools have been recently developed and showcase great precision when it comes to detecting and segmenting stomata. They utilize a wide range of techniques, including deep residual networks, Mask R-CNN, and YOLO-based models. Every tool has its own set of distinctive features that target specific research requirements. These include the ability to handle noisy images, generate comprehensive statistical outputs, and offer user-friendly web interfaces.

Feature	LeafNet	StomaAI	StomataCounter	StomaVision
Core algorithm	Deep learning (StomaNet, LeafSeg)	Mask R-CNN, geometric calculations	DCNN, fully convolutional network	YOLOv7, VDP
Image segmentation	Yes	Yes	Yes	Yes
Stomatal count	Yes	Yes	Yes	Yes
User interface	Software	Software	Web Service	Web Platform/Docker
Unique features	Hierarchical strategy	Detailed geometric calculations	High accuracy, fully automated	Web-based, rotated bounding boxes

Table 1: Comparison of tools

In addition, there are other stomata detection tools that are currently available. Systems like StoManager1[20], StomataScorer[27], LabelStoma[28], and DeepStomata[29] also provide comprehensive stomatal analysis; however, their fundamental techniques are closely aligned with the ones described before. For example, both StoManager1 and LabelStoma use YOLO models, which are a type of convolutional neural network, although they employ distinct versions. Similarly, systems like DeepStomata and StomataScorer use convolutional neural networks to analyze the shape and structure of stomata, similar to the approaches used by LeafNet and StomaAI. Therefore, only four approaches, including LeafNet, StomaAI, StomataCounter, and StomaVision, were covered.

3 The aims of the thesis

Although there are many advanced tools for stomatal detection and segmentation, there are certain challenges that make them unsuitable to my research. These challenges are primarily related to the nature of the samples I am working with, the type of data (videos rather than images), and the quality of the images obtained from bright-field microscopy on nail varnish leaf impressions.

Working with video data captured across different focal planes presents a significant difficulty. Unlike static images, video data requires a different approach. Current available tools like LeafNet, StomataCounter, StomataAi and others, are specifically created to analyze high-quality static images of plant leaf epidermis and do not consider the variability introduced by video data.

The aims of my thesis are:

- To create an algorithm for detection and counting of stomata on a single frames from the videos.

- Combine data from the frames to analyse overall video and detect stomata.

4 Experimental part

4.1 Materials and methods

For the first two methods in next section, I used different frames extracted from the microscopic videos with stomata.

4.1.1 First approach: adaptive thresholding and blob detection

The initial method that I used to detect stomata on a sigle frame (image) from the video was interactive thresholding and blob detection. I was manually adjusting the threshold value using a track bar to distinguish stomata from the pavement cells. By tuning threshold, I aimed to convert the grayscale image into a binary image where stomata would appear as distinct black blobs against a white background. However, due to similar coloration of pavement cells and stomata, it is impossible to do ideally, meaning that some pavement cells also will appear as black.

After determining the optimal threshold value that separated some stomata, I applied blob detection. Blob detection is a technique that finds regions in an image that differ in brightness and color. I configured the blob detector with parameters to detect blobs filtered by area, circularity, inertia, and convexity. These filters helped refine the detection by focusing on blobs with specific sizes and shapes, thereby filtering out pavement cells and making blob detection more precise. The algorithm identified key points, which represent the center and size of each detected blob. Then I

used these key points to draw rectangles around each blob, giving a visual indication of their location.

4.1.1.1 Result of the 1st method

Here I provide the detection results using the interactive thresholding and blob detection approach. On Figure 10, blob detection is displayed on the thresholded image, and on the original image in Figure 11.

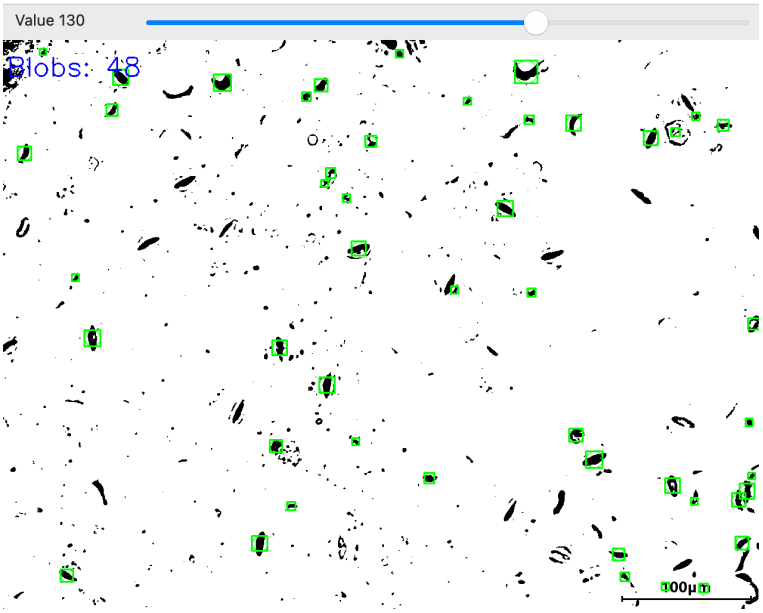

Figure 10: Thresholded image with applied blob detection.

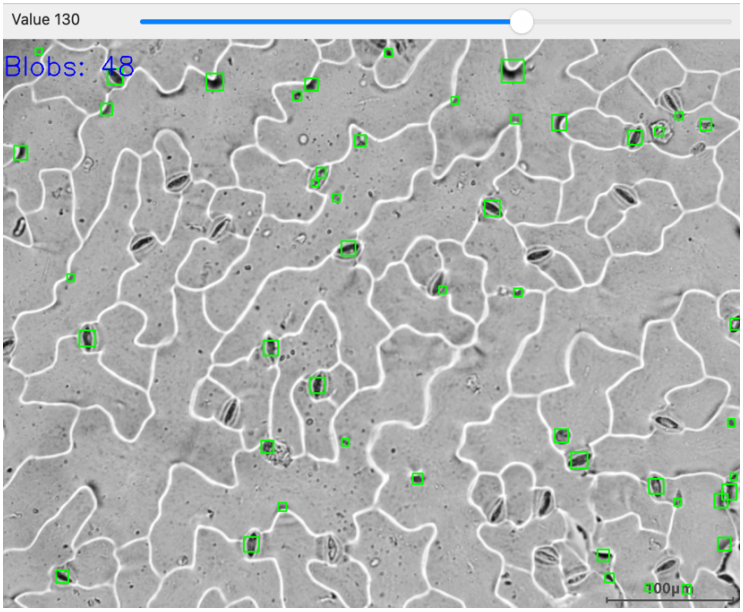

Figure 11: Original image with applied blob detection.

By combining the interactive nature of the thresholding process with blob detection, I was able to interactively move a trackbar and immediately see the results of thresholding and blob detection. The algorithm continuously updated the count of detected blobs while displayed on the output image.

This approach was not successful, I encountered difficulties in perfectly isolating the stomata, as the blob detection often identified pavement cells alongside stomata.

4.1.2 Second approach: 2d FFT and band-pass filter

To improve stomatal detection on the image detection, I moved to using two-dimensional Fast Fourier Transform (2D FFT) and band-pass filtering. Two-dimensional Fast Fourier Transform (2d FFT) and band-pass filtering concentrate on modifications of frequencies of image pixel intensities to enhance the detection of features within an image, such as stomata. This method is more general than previous one, meaning it can detect some stomata on different images without adjusting the parameters.

Initially, I used 2D FFT to transform my image from the spatial domain to the frequency domain representation. This helped to transform the image into its frequency components, representing the image in terms of its sinusoidal patterns. By shifting the zero-frequency component to the center of the spectrum, I obtained a magnitude spectrum, which visually highlights the frequency components of the image. This visualization represents the intensity of various frequencies present in the image (Figure 12). The bright center indicates the prevalence of low-frequency components, which are typical of smoother regions and broader patterns in the image. The faint lines extending from the center suggest some directional features, which are the stomata. This step is crucial for understanding and manipulating the image based on its frequency components.

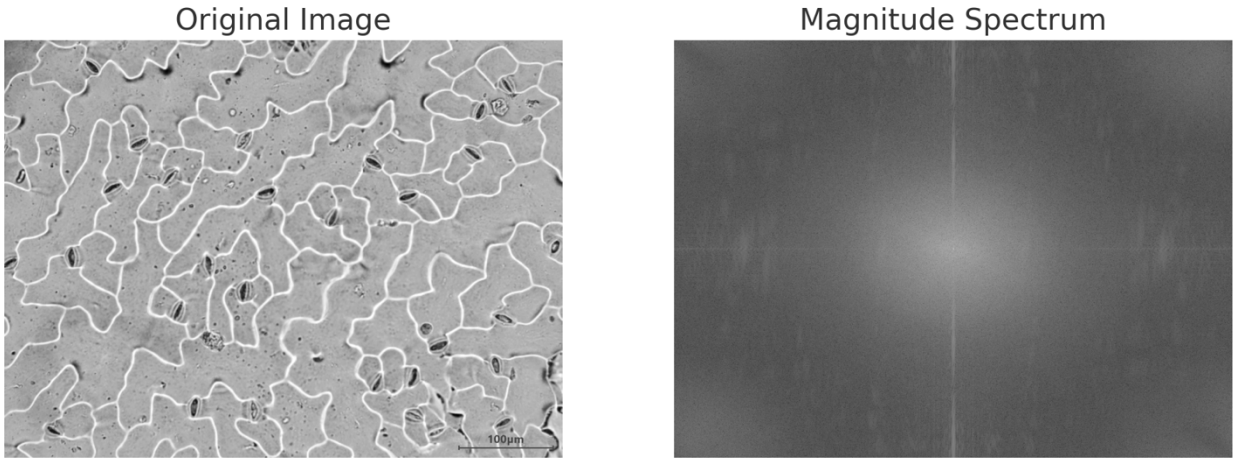

Figure 12: Original image and its corresponding magnitude spectrum obtained using the 2D Fast Fourier Transform (FFT)

After obtaining image in frequency domain, I used band-pass filter to allow frequencies within a specific range to pass while cutting frequencies outside this range. The parameters of the band-pass filter that I used were cutoff distance from the center (d) and the filter order (n). Cutoff distance from the center (d) defines the range of frequencies around the center of the frequency spectrum (representing low frequencies) that the filter allows to pass. It isolates the frequencies corresponding to the stomata's size and spatial distribution. The filter order (n) controls how quickly the filter transitions from allowing certain frequencies to blocking others. Together they control the range of frequencies that are allowed to pass through the filter.

After isolation of frequency components that passed through and returning image to the spatial domain (Figure 13), I applied thresholding to create a binary image where the stomata appeared as white regions against a black background (Figure 14).

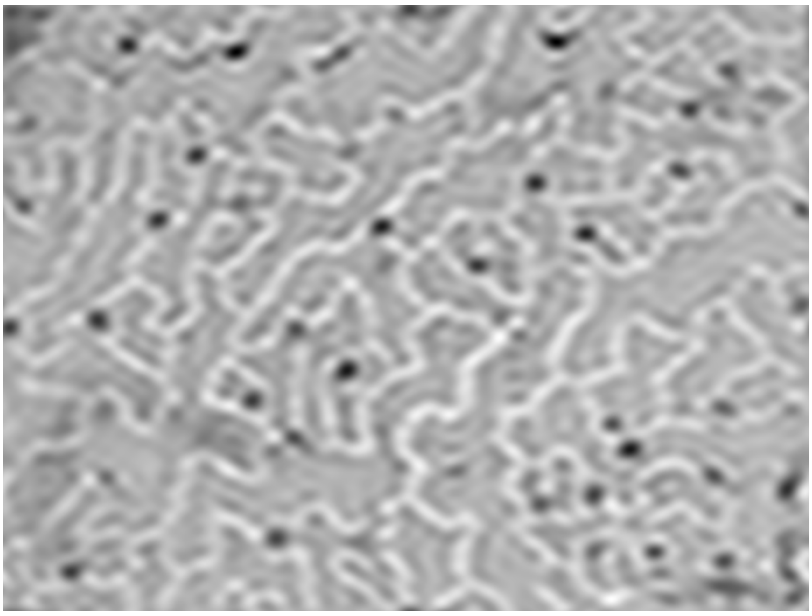

Figure 13: Frequency filtered image

Figure 14: Thresholded image of isolated stomata after 2d FFT and band-pass filter

To identify and count the stomata, I used contour detection to draw contour boundaries of the detected objects and in order to ensure accuracy, I implemented additional checks for circularity and area, filtering out non-stomatal features that might have similar shapes but did not meet the criteria for stomata.

4.1.2.1 Result of the 2nd method

The contours were drawn on the original image for visualization (Figure 15).

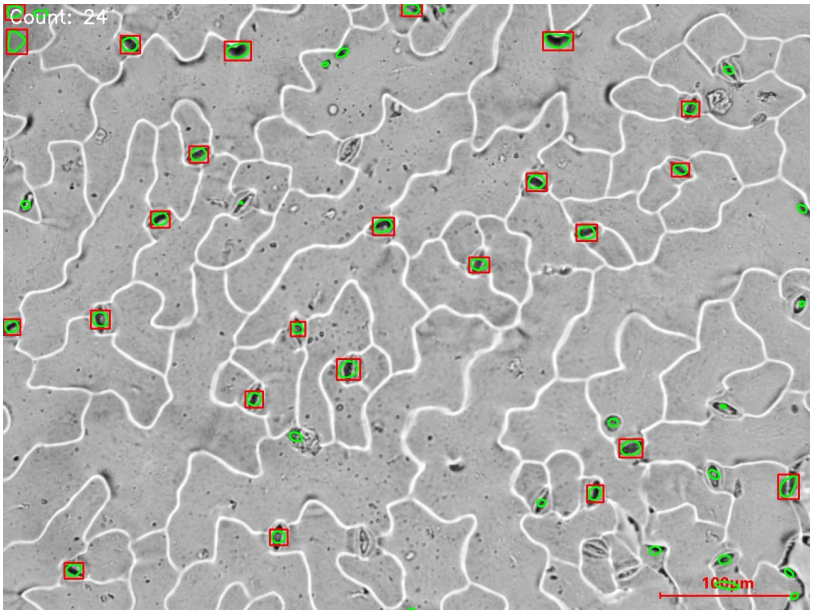

Figure 15: Detected stomata on a single frame from the video using image processing pipeline

The challenges with this method included adjusting the band-pass filter's settings and making sure that the filter successfully separated the stomatal frequencies without losing a lot of information. Another challenge was dealing with varying image quality and ensuring that the contour detection was robust enough to handle different lighting conditions and noise levels.

After treying this approach on differen images, I came to the conclusion that this approach requires constant adjustments of the thresholding value to detect stomata on different frames, which makes it a non-general method, meaning that it is not applicable to different frames without manual parameter adjustments.

4.1.3 Third approach: YOLOv8 detection on images

While image processing techniques like interactive thresholding, blob detection, and FFT with band-pass filtering provided some success in detecting stomata, they required significant manual adjustments and were not fully effective for automated detection in video frames. This led to the exploration of more advanced techniques such as using the YOLOv8 model for automated stomata detection.

4.1.3.1 Data configuration

I created a dataset of images with labelled stomata from 80 microscopic videos. Since stomata come into clear view as the focal plane moves up and down, each frame of the video has different focus, meaning stomata are visible differently in each frame of every video. To be sure that I have all stomata from each video nicely visible, I took 2-5 frames from each video and manually labelled stomata using Make Sense Ai software [29] as a tool. Those labels were further used for training a model for stomatal detection.

Dataset was divided into 3 subsets: **Training set, Test set and Validation set**. This split is important because it allows us to evaluate the model's performance on new, unseen images to ensure it can generalize well and not just memorize the training images.

My training set consisted of training data that included 200 images (input data) and corresponding labels for these images, which indicate the locations of stomata and what they look like. These images and labels were used to teach the model what stomata look like and where the stomata are located during model training.

I also used data augmentation to increase the size of the training data set and consequently the efficiency of learning, as more data leads to better learning, so the model will better detect stomata. Data augmentation artificially increases the size of training dataset by applying various transformations to original images. This helped the model to generalize better and reduced the risk of overfitting.

I used the following data augmentation techniques:

- Rotation: rotates the images by a certain angle;
- Scaling: zoom the images in or out;
- Translation: shifts the images horizontally or vertically;
- Flipping: flips the images horizontally or vertically;
- Brightness adjustment: increases or decreases brightness of the images;
- Contrast adjustment: changes the contrast of the images;
- Noise addition: adds random noise to the images.

My test set, in contrast, consisted of validation data that included 50 images and corresponding labels for these images. These images and labels were different from the training data set images and labels. Test data that is different from training data was used to evaluate the performance of my model during training by monitoring how well the model is working on unseen data. The validation set helps in detecting overfitting. If the model performs well on the training set but poorly on the test set, it indicates overfitting, meaning that model learns the training data too well, including the noise and details that are not important instead of learning general features like overall stomatal characteristics.

Lastly, my validation set consisted of completely new data that were images model had not seen before, they were different from images in training and test sets and used to check how well the model has learned to detect stomata after training is complete. The test set does not influence the training process in any way.

4.1.3.2 Transfer learning for YOLOv8

Training process started with providing the model with the training data of 200 images where each image came with labels that indicate where the stomata are located and how they look like. Then model created random "weights", the parameters it fine-tuned during training to make accurate predictions. The model looks at an image, makes an initial guess about how stomata look like and where stomata are, and compares its guess to the actual stomatal labels. So, the model makes predictions. After that, the model calculates how wrong its guess was. This difference is called the "error." Based on the error, the model adjusts its weights many times to reduce the error for each image and over time gets better and better at detecting stomata. By the end of the training, the weights were tuned so that the model could accurately detect stomata in new images. This process

was repeated for many epochs (iterations over the dataset) and after training, the model was saved to a file (best.pt) that contained the trained weights.

4.1.3.3 Results for the 3d method

For evaluating the model, I applied the pre-trained YOLOv8 model to detect stomata in the validation set and then on completely new images of the test set. This step helped me to understand if the model can generalize its learning or if it only works well on the training data.

Figure 16 shows the result after applying my pre-trained YOLOv8 model on the image from the validation data:

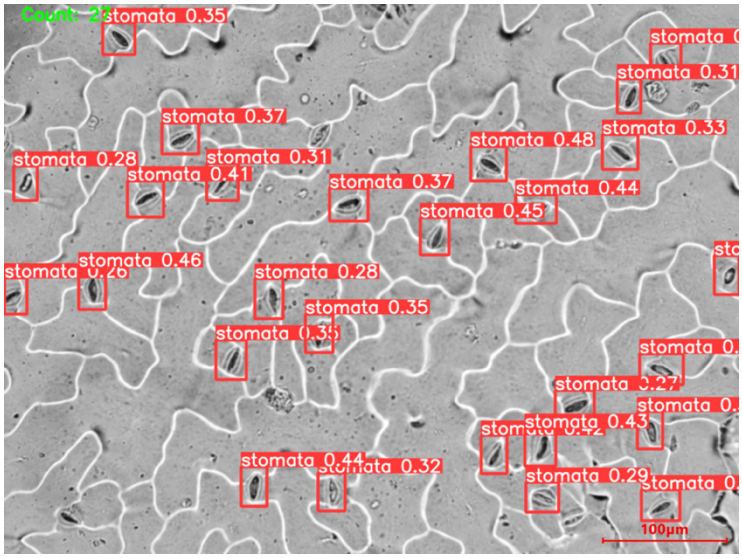

Figure 16: Detected and counted stomata on a validation data single frame from the video

Figure 17 shows the result of applying my model on a completely new image, that the model has not seen before (from the test data set):

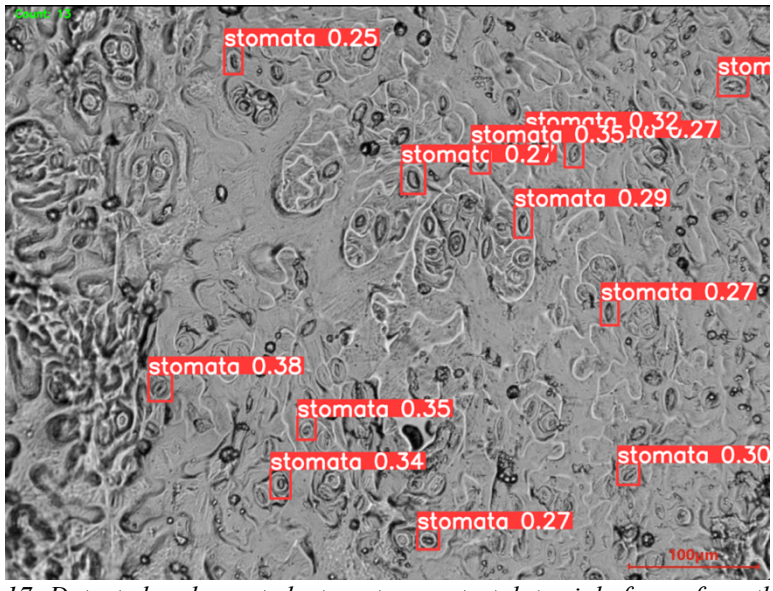

Figure 17: Detected and counted stomata on a test data sigle frame from the video

Bounding boxes (rectangles) were used to specify the detected stomata and by calculating the amount of boxes, I obtained the stomatal count.The numbers displayed under the bounding boxes represent the confidence scores of the detections made by the model. These scores indicate the model's confidence that the detected object is indeed a stoma.

At this stage, I discovered a little disadvantage of my model that potentially decreased its robustness and acuraccy on the test data. Some of the images that I used for test data were not completely different from the training images. As stated before, I extracted 2–5 frames from each video to create a dataset, so some of the images in the training data and validation data were frames from the same videos.

4.1.4 Fourth approach: YOLOv8 detection on videos

In this approach, we were not just detecting stomata in a single static image, but continuously throughout an entire video capturing different focal planes. The process involved running my YOLOv8 model on every single frame of each video to detect stomata. For each detected stoma, we extracted the x and y coordinates of its location and used them to draw a circle at the exact same location on a black image that was the same size as the frame. This was done for each frame, creating individual black images that represent the presence of stomata at specific locations for each frame.

Next, individual black images were accumulated into one single black image of the same size, creating a heatmap, which is the visual representation of how frequently stomata were detected on the frames of the video during detection process and where stomata were detected. Stomata detected multiple times (e.g., 10 times) appeared as brighter circles on the accumulated black image then stomata that were detected less frequently. Essentially, we were layering the circles corresponding to stomata on top of each other. The brighter the spot, the more frequently stomata were detected at that location.

```python
# Creating an empty black image for accumulation(heatmap)
heatmap = np.zeros((frame_height, frame_width), dtype=np.float32)

while cap.isOpened():
    ret, frame = cap.read()
    if not ret:
        break

    # running model through each frame
    results = model.predict(frame)

    # createing a temporary image to hold the current frame's detections
    temp_image = np.zeros((frame_height, frame_width), dtype=np.float32)

    # extracting coordinares of stomata
    for box in results[0].boxes.xyxy:
        x1, y1, x2, y2 = map(int, box[:4])
        # center of the bounding box
        center_x = (x1 + x2) // 2
        center_y = (y1 + y2) // 2
        # circle at the center of the bounding box on the temporary image
        cv2.circle(temp_image, (center_x, center_y), radius=20, color=1, thickness=-1)

    # adding the temporary image to the heatmap
    heatmap += temp_image
```

Figure 18: The processing pipeline for creating a heatmap

Results

Stomatal detection on videos results in a black image with white spots that indicate the locations of stomata. The brighter the spot, the more frequently stomata were detected at that location during video analysis. For displaying the result, I used video, which contains frames from validation data, so the model has not seen them before (Figure 19).

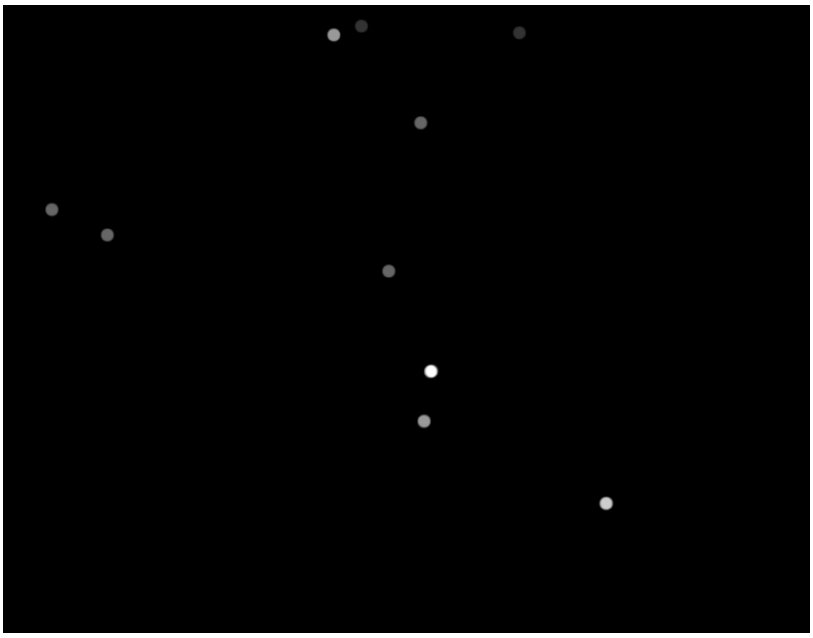

Figure 19: Heatmap that shows the frequency of stomata detections and their locations

The stomatal count was achieved by drawing a contours around the white spots (Figure 20).

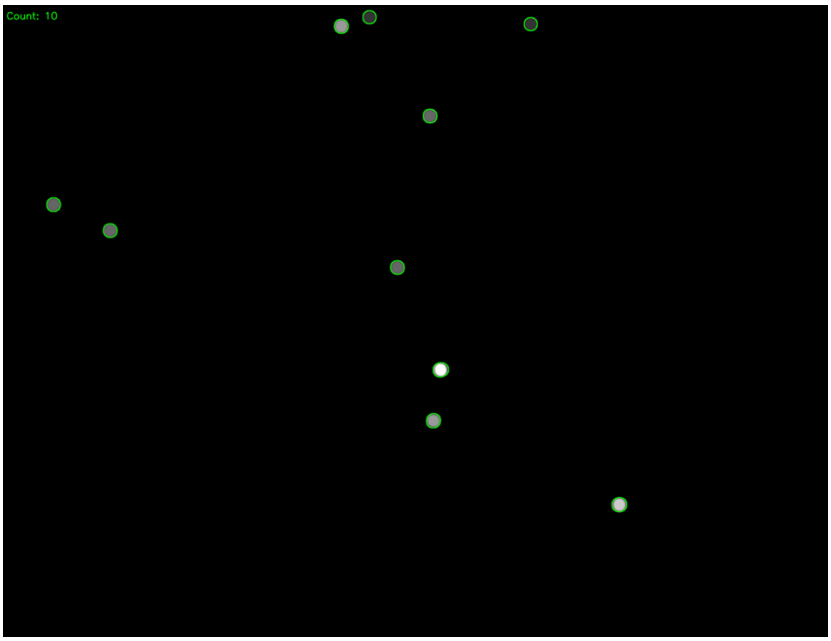

Figure 20: Heatpmap with known stomatal count.

Additionally, I applied the YOLOv8 stomatal detection model on the videos, frames of which were used in the training and test set, because I knew stomatal density values for these videos.

Here is the comparison table:

Sample number	Stomatal count	BY YOLO MODEL
1A	35	16
1Y	30	19
2A	34	31
2Y	25	23
3A	33	32
3Y	27	16
4A	35	31
4Y	31	25
5A	35	29
5Y	26	23
6A	42	35
6Y	38	37
7A	36	32
7Y	32	28
8A	54	47
8Y	35	32
9A	33	31
9Y	19	15
10A	30	24
10Y	21	18

Based on the comparison table of stomatal counts by the YOLOv8 model and the actual stomatal counts, I can state that the YOLOv8 model generally captures the trends in stomatal counts across different samples. For most samples, the model's predictions are relatively close to the actual counts, however, they are not equal, and on some samples, the detection appeared imprecise.

The described approach allowed us to combine frames from a video into a single compressed image, showing not only the locations of stomata but also how frequently the model detected stomata at each location. Even if the model did not detect a stoma in some frames, it could still be detected in other frames, as the stomatal locations remain consistent throughout the video.

Discussion

This thesis investigates the application of machine learning and deep learning approaches for automatic detection and measurement of stomatal cells. The developed algorithm primarily used the YOLOv8 model to analyze video frames of leaf epidermal impressions and detect stomata. The results indicate that the YOLOv8 model is highly effective in detecting stomata acrossdifferent frames of video data, even when the stomata appear at different focal depths.

To enhance both the accuracy and reliability of the model for detecting stomata, it is essential to create a more distinct and diverse test set. To achieve a more comprehensive evaluation of the model's performance on the videos and more accurate evaluation of its generalization capabilities, it is crucial to ensure that the test images are completely distinct from the training images. The images should come from completely different microscopic videos than the ones in the test data.

Additionally, further enhancing the model's learning process can be achieved through adding an extra number of frames and stomata labels into the training data . Increasing the size and diversity of the training set allows the model to acquire a broader range of knowledge about stomata, resulting in enhanced accuracy in detecting them.

Implementing the YOLOv8 model for stomata detection presents new opportunities for automated plant phenotyping. Efficient analysis of video data can greatly simplify studies in plant physiology and enhance result consistency by reducing human error.

Summary

The aim of this thesis was to develop an algorithm for the automated detection and quantification of stomatal cells using video data of leaf epidermal imprints. By employing the YOLOv8 model, I achieved significant improvements in detection accuracy and processing speed compared to manual methods. The results demonstrated the model's effectiveness in handling varying focal depths within video frames, providing consistent stomatal counts.

Future enhancements can include retraining by focusing on two aspects: enhancing the validation set to make it more distinct, and increasing the amount of training data. Implementing these methods will likely result in a stronger and more precise model for stomatal detection from videos.

Bibliography

[1] C. A. Schneider, W. S. Rasband, and K. W. Eliceiri, “NIH Image to ImageJ: 25 years of image analysis,” *Nat Methods*, vol. 9, no. 7, pp. 671–675, Jul. 2012, doi: 10.1038/nmeth.2089.

[2] N. Sai *et al.*, “StomaAI: an efficient and user-friendly tool for measurement of stomatal pores and density using deep computer vision,” *New Phytologist*, vol. 238, no. 2, pp. 904–915, Apr. 2023, doi: 10.1111/nph.18765.

[3] T. Yoshikura, “What Is Brightfield Microscopy?” Accessed: May 13, 2024. [Online]. Available: https://www.olympus-lifescience.com/en/discovery/what-is-brightfield-microscopy/

[4] S. A. Casson, K. A. Franklin, J. E. Gray, C. S. Grierson, G. C. Whitelam, and A. M. Hetherington, “phytochrome B and *PIF4* Regulate Stomatal Development in Response to Light Quantity,” *Current Biology*, vol. 19, no. 3, pp. 229–234, Feb. 2009, doi: 10.1016/j.cub.2008.12.046.

[5] S. Chakraborty, M. Roy, and S. Hore, “A Study on Different Edge Detection Techniques in Digital Image Processing,” in *Feature Detectors and Motion Detection in Video Processing*, IGI Global, 2017, pp. 100–122. doi: 10.4018/978-1-5225-1025-3.ch005.

[6] N. Fahlgren, M. A. Gehan, and I. Baxter, “Lights, camera, action: high-throughput plant phenotyping is ready for a close-up,” *Current opinion in plant biology*, vol. 24, pp. 93–99, 2015.

[7] “Ludwig_ImageConvolution.pdf.” Accessed: May 23, 2024. [Online]. Available: https://web.pdx.edu/~jduh/courses/Archive/geog481w07/Students/Ludwig_ImageConvolution.pdf

[8] A. Hartmann, T. Czauderna, R. Hoffmann, N. Stein, and F. Schreiber, “HTPheno: An image analysis pipeline for high-throughput plant phenotyping,” *BMC Bioinformatics*, vol. 12, no. 1, p. 148, Dec. 2011, doi: 10.1186/1471-2105-12-148.

[9] M. M. Rahaman, D. Chen, Z. Gillani, C. Klukas, and M. Chen, “Advanced phenotyping and phenotype data analysis for the study of plant growth and development,” *Front. Plant Sci.*, vol. 6, Aug. 2015, doi: 10.3389/fpls.2015.00619.

[10] D. Leister, C. Varotto, P. Pesaresi, A. Niwergall, and F. Salamini, “Large-scale evaluation of plant growth in Arabidopsis thaliana by non-invasive image analysis,” *Plant Physiology and Biochemistry*, vol. 37, no. 9, pp. 671–678, 1999.

[11] A. Walter, F. Liebisch, and A. Hund, “Plant phenotyping: from bean weighing to image analysis,” *Plant Methods*, vol. 11, no. 1, p. 14, 2015, doi: 10.1186/s13007-015-0056-8.

[12] M. T. Campbell, A. C. Knecht, B. Berger, C. J. Brien, D. Wang, and H. Walia, “Integrating Image-Based Phenomics and Association Analysis to Dissect the Genetic Architecture of Temporal Salinity Responses in Rice,” *Plant Physiology*, vol. 168, no. 4, pp. 1476–1489, Aug. 2015, doi: 10.1104/pp.15.00450.

[13] J. De Vylder, F. Vandenbussche, Y. Hu, W. Philips, and D. Van Der Straeten, “Rosette Tracker: An Open Source Image Analysis Tool for Automatic Quantification of Genotype Effects,” *Plant Physiology*, vol. 160, no. 3, pp. 1149–1159, Nov. 2012, doi: 10.1104/pp.112.202762.

[14] J.-M. Pape and C. Klukas, “3-D Histogram-Based Segmentation and Leaf Detection for Rosette Plants,” in *Computer Vision - ECCV 2014 Workshops*, vol. 8928, L. Agapito, M. M. Bronstein, and C. Rother, Eds., in Lecture Notes in Computer Science, vol. 8928. , Cham: Springer International Publishing, 2015, pp. 61–74. doi: 10.1007/978-3-319-16220-1_5.

[15] M. Minervini, M. M. Abdelsamea, and S. A. Tsaftaris, “Image-based plant phenotyping with incremental learning and active contours,” *Ecological Informatics*, vol. 23, pp. 35–48, Sep. 2014, doi: 10.1016/j.ecoinf.2013.07.004.

[16] S. A. Tsaftaris, M. Minervini, and H. Scharr, “Machine learning for plant phenotyping needs image processing,” *Trends in plant science*, vol. 21, no. 12, pp. 989–991, 2016.

[17] O. Simeone, “A Very Brief Introduction to Machine Learning With Applications to Communication Systems,” *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 4, pp. 648–664, Dec. 2018, doi: 10.1109/TCCN.2018.2881442.

[18] K. O’Shea and R. Nash, “An Introduction to Convolutional Neural Networks.” arXiv, Dec. 02, 2015. doi: 10.48550/arXiv.1511.08458.

[19] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015, doi: 10.1038/nature14539.

[20] J. Wang, H. J. Renninger, Q. Ma, and S. Jin, “Measuring stomatal and guard cell metrics for plant physiology and growth using StoManager1,” *Plant Physiology*, vol. 195, no. 1, pp. 378–394, May 2024, doi: 10.1093/plphys/kiae049.

[21] J. Redmon and A. Farhadi, “YOLO9000: Better, Faster, Stronger.” arXiv, Dec. 25, 2016. doi: 10.48550/arXiv.1612.08242.

[22] “How to train and use a custom YOLOv7 model,” Paperspace Blog. Accessed: May 16, 2024. [Online]. Available: https://blog.paperspace.com/yolov7/

[23] Z. Huang, L. Li, G. C. Krizek, and L. Sun, “Research on Traffic Sign Detection Based on Improved YOLOv8,” *Journal of Computer and Communications*, vol. 11, no. 7, Art. no. 7, Jul. 2023, doi: 10.4236/jcc.2023.117014.

[24] S. Li *et al.*, “LeafNet: a tool for segmenting and quantifying stomata and pavement cells,” *The Plant Cell*, vol. 34, no. 4, pp. 1171–1188, 2022.

[25] K. C. Fetter, S. Eberhardt, R. S. Barclay, S. Wing, and S. R. Keller, “StomataCounter: a neural network for automatic stomata identification and counting,” *New Phytologist*, vol. 223, no. 3, pp. 1671–1681, Aug. 2019, doi: 10.1111/nph.15892.

[26] T.-L. Wu *et al.*, “StomaVision: stomatal trait analysis through deep learning.” bioRxiv, p. 2024.04.24.590919, Apr. 28, 2024. doi: 10.1101/2024.04.24.590919.

[27] X. Liang *et al.*, “StomataScorer: a portable and high-throughput leaf stomata trait scorer combined with deep learning and an improved CV model,” *Plant Biotechnology Journal*, vol. 20, no. 3, pp. 577–591, 2022, doi: 10.1111/pbi.13741.

[28] A. Casado-García *et al.*, “LabelStoma: A tool for stomata detection based on the YOLO algorithm,” *Computers and Electronics in Agriculture*, vol. 178, p. 105751, Nov. 2020, doi: 10.1016/j.compag.2020.105751.

[29] Y. Toda, S. Toh, G. Bourdais, S. Robatzek, D. Maclean, and T. Kinoshita, “DeepStomata: Facial Recognition Technology for Automated Stomatal Aperture Measurement.” bioRxiv, p. 365098, Jul. 09, 2018. doi: 10.1101/365098.

Appendices

Here is the link to the repository that includes different python codes directly relavant to my thesis:

https://github.com/ivangorbachenko/Thesis_Gorbachenko

NON-EXCLUSIVE LICENCE TO REPRODUCE THESIS AND MAKE THESIS PUBLIC

I, Ivan Gorbachenko
1. herewith grant the University of Tartu a free permit (non-exclusive licence) to reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright,

“Detection and Quantification of Stomata”,

supervised by Hanna Hõrak and co-supervised by Rain Eric Haamer

2. I grant the University of Tartu a permit to make the work specified in p. 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons licence CC BY NC ND 3.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work until the expiry of the term of copyright.

3. I am aware of the fact that the author retains the rights specified in p. 1 and 2.

4. I certify that granting the non-exclusive licence does not infringe other persons’ intellectual property rights or rights arising from the personal data protection legislation.

Ivan Gorbachenko

22/05/2024