

Innovative Leaf Disease Mapping: Unsupervised Anomaly Detection for Precise Area Estimation

Lucas Chrisph Johannsen

Hochschule Osnabrück

Ameer Tamoor Khan

atk@plen.ku.dk

University of Copenhagen

Signe Marie Jensen

University of Copenhagen

Jochen Kruppa-Scheetz

Hochschule Osnabrück

Research Article

Keywords: Anomaly Detection, Unsupervised, Leaf Infection, Computer Vision

Posted Date: August 21st, 2024

DOI: <https://doi.org/10.21203/rs.3.rs-4797098/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

Innovative Leaf Disease Mapping: Unsupervised Anomaly Detection for Precise Area Estimation

Lucas Chrisph Johannsen
Hochschule Osnabrück
University of Applied Sciences
Lucasjohannsen@gmail.com

Ameer Tamoor Khan
University of Copenhagen
atk@plen.ku.dk

Signe Marie Jensen
University of Copenhagen
smj@plen.ku.dk

Jochen Kruppa-Scheetz
Hochschule Osnabrück
University of Applied Sciences
j.kruppa@hs-osnabrueck.de

July 24, 2024

Abstract

Detecting and quantifying the diseased regions in a leaf is an important task in plant breeding in order to select plants based on disease resistance. Ratings done by eye and hand are not accurate, can be highly subjective and take a lot of work hours. Using machine learning, it is possible to generate faster more accurate data. By combining modern anomaly detection algorithms with masking algorithms a robust method capable of estimating the infected leaf area was developed, resulting in a novel method with superior results. Using unsupervised models both for the masking and for the detection, the method can easily be used for different kinds of detection tasks.

Keywords

Anomaly Detection, Unsupervised, Leaf Infection, Computer Vision

1 Introduction

Quantifying disease resistance is crucial for the selection of disease-resistant plants in breeding programs. Disease resistance is often governed by multiple genes, resulting

in a spectrum of resistance levels that range from highly resistant to highly susceptible plants. Therefore, accurately measuring or evaluating the severity of diseases is of paramount importance [22]. Traditionally, direct measurements of disease severity have involved quantifying the affected area on plant organs, such as leaves or stems. This also includes assessing parameters that are indirectly influenced by diseases, such as crop yield and quality [17].

The conventional approach to estimating the extent of diseased areas on plant leaves relies heavily on subjective visual assessments. This process requires extensive training to ensure the accuracy and consistency of the evaluations [7]. Typically, ordinal scales ranging from 1 (indicating very little disease presence) to 9 (indicating very severe disease presence) are employed, which are similar to the Likert scale used in social sciences. These scales help to reduce biases such as the raters' preference for certain percentage values like 5, 10, or 50 [5, 13]. However, despite these measures, manual ratings are inherently subject to variability over time and between different raters, which can lead to inconsistent and unreliable data.

This inherent variability raises a critical question: *Can unsupervised machine learning techniques effectively replace manual ratings in the evaluation of disease severity?* Addressing this question is essential for improving

the precision and objectivity of disease assessments in plant breeding.

To meet the need for precise estimations of the diseased leaf area in plant breeding, this work proposes a novel method based on machine learning. This method adopts an unsupervised approach, which offers greater flexibility and applicability across a wide range of crops and phenotypes. Specifically, the method combines an unsupervised anomaly detection model with a leaf selection algorithm to improve the accuracy of estimating diseased areas on plant leaves [22].

The implementation of this method is available in the [Diffusion Diseases Repository](#). This repository provides the necessary resources and guidelines for applying the proposed machine learning approach to different plant species and disease phenotypes, facilitating broader adoption in the field of plant pathology and breeding.

2 Related Works

Accurate detection of plant diseases is a critical prerequisite for precise estimation of the affected area. Recent advancements in computer vision, particularly through the application of deep learning models, have significantly enhanced the capabilities for detecting plant diseases [26, 6]. It is important to differentiate between image-level predictions, commonly known as image classification, and pixel-level predictions, referred to as image segmentation. While image classification is adept at distinguishing between healthy and diseased plants, image segmentation provides the detailed information necessary for accurate estimation of the disease-affected area [24, 11, 16].

Supervised learning models, which are widely used in these applications, require pre-labeled ground truth masks. The process of generating these labels is both time-consuming and costly [34, 31]. Unsupervised machine learning, in contrast, does not rely on annotated data. Instead, it employs various learning methods to detect disease by treating it as an anomaly compared to healthy plant tissue. This approach has been explored extensively in both plant disease detection and medical imaging, where anomaly detection models are tailored to meet specific needs such as rapid inference times and the ability to handle diverse types of anomalies [19, 21, 1].

Innovative unsupervised methods for plant disease detection have been explored, including the use of autoencoders and encoder-decoder networks to detect anomalies based on color differences between healthy and diseased leaf images [2, 10]. These models identify deviations from the norm by analyzing patterns in the data. To estimate the percentage of the diseased area on leaves, leaf segmentation algorithms are employed. Both supervised and unsupervised approaches to background-foreground segmentation are considered, with unsupervised methods generally performing more slowly and less effectively than their supervised counterparts [30, 35].

In the context of plant disease detection, unsupervised models have the potential to transform the way diseases are identified and quantified. These models, which have been successfully developed for industrial defect or anomaly detection, can be adapted to plant disease scenarios. One notable model is the Revisiting Reverse Distillation (RD++) developed by Tien et al. [29]. RD++ enhances the Reversed Distillation (RD) approach [8] and has achieved the highest Area Under the Receiver Operating Characteristic (AUROC) score on the MVTEC dataset [19] at the time of its selection.

The predecessor of the RD++ model, the RD method, utilizes a teacher-student model architecture. In this setup, a pre-trained teacher encoder remains frozen and is capable of extracting distinct features from the data. This allows the student model to learn by mimicking the teacher’s output, which is particularly effective for anomaly detection where labeled data is scarce. Applying such a model to the detection of plant diseases is expected to yield high performance, offering a robust solution for precise disease area estimation in various agricultural applications.

3 Proposed Method

The training process, illustrated in Figure 1, involves a trainable student decoder that learns to replicate the teacher’s representations when trained on data devoid of anomalies. The RD++ method introduces modifications to the RD architecture and loss function. By incorporating pseudo-anomaly noise during training, the compact feature representations are trained to suppress abnormal information, thereby preventing the student network from

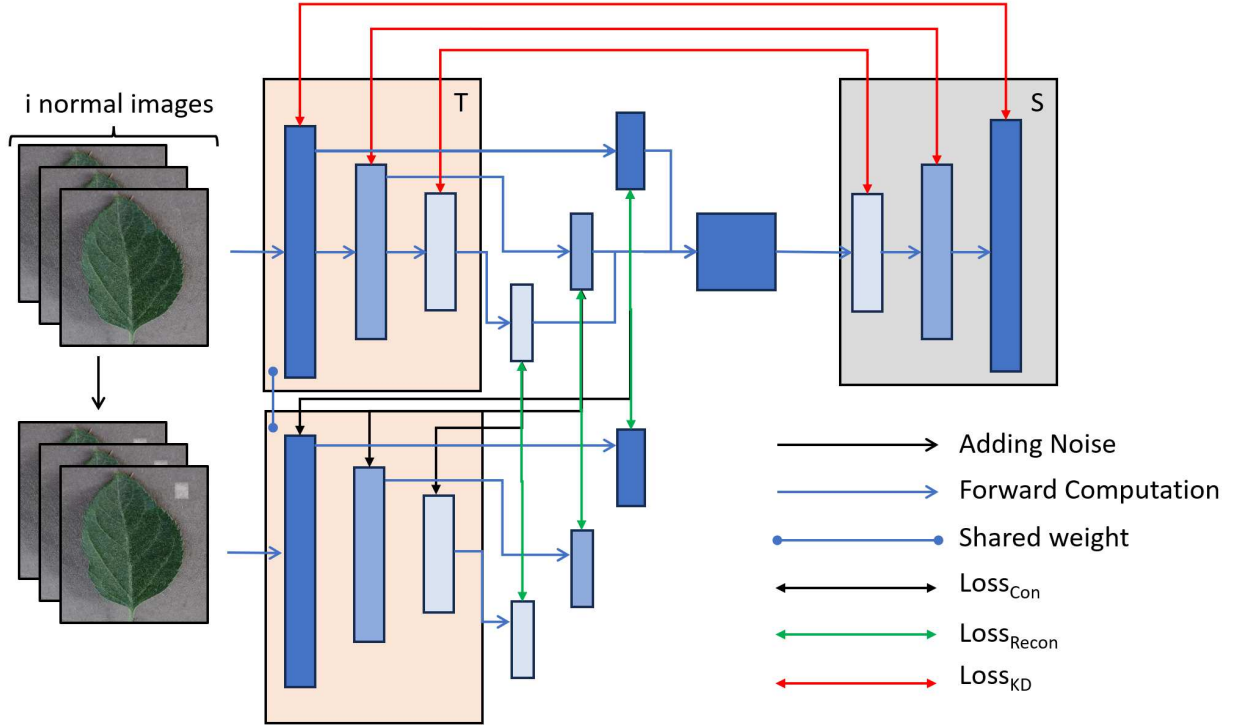


Figure 1: Training Process for the RD++. The input images without anomalies are fed with and without noise into the teacher network (T). The information is then fed forward within the teacher network to compact feature presentation before fed into the OCBE module. Then the student network (S) replicates the input for the OCBE module. A combination of loss functions is optimised during training.

receiving such information. Pseudo-anomalies are simulated using Simplex noise [32] during the training phase. Additionally, a projection layer is introduced to receive features from the corresponding teacher’s block. These projection layers, encouraged by a new reconstruction loss, learn to reconstruct normal features during training. A contrastive loss ensures a focus on deeper representations of normal features by excluding abnormal information. Moreover, a self-supervised optimal transport loss is employed to minimize the distance between feature embeddings in the compact feature space.

During inference, the advanced RD++ anomaly detection model performs a series of steps to detect diseased regions across various levels of complexity, as depicted in Figure 2. The complexity of features is associated with

the depth of the respective layer in the model’s architecture.

The inference image is first processed by the teacher network, where it undergoes manipulation to generate three representations with increasing feature complexity. Shallow layers extract low-level features such as color and texture, while high-level layers capture more global information. These representations are passed to their corresponding compact feature presentations, which are trained to suppress abnormal features and compact them. The three compact feature presentations are then fed into the OCBE module, which further compacts the inputs. Subsequently, the student network reconstructs the input with decreasing feature complexity and increasing spatial resolution. The cosine similarity between the teacher and

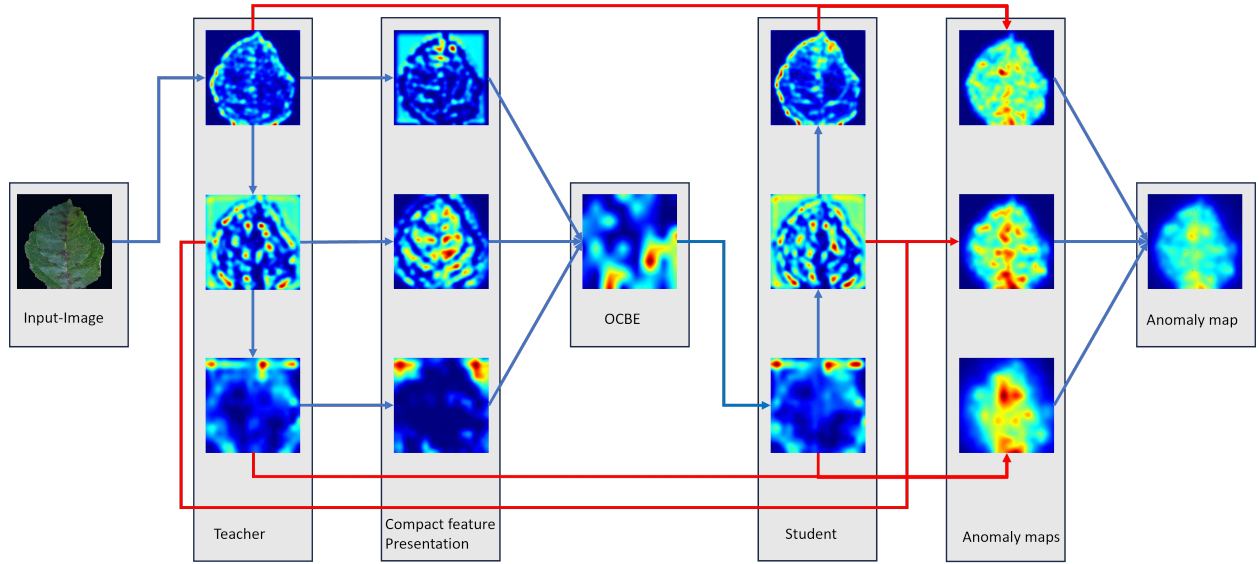


Figure 2: The training process for the RD++ model involves feeding input images, both with and without noise, into the Teacher network (T). The Teacher network processes these images into a compact feature representation, which is then fed into the OCBE module. The Student network (S) replicates this input for the OCBE module. During training, a combination of loss functions is optimized to improve the model’s performance.

the corresponding student layers is calculated, with differences indicating anomalies due to the student’s inability to restore abnormal features.

The complexity of features and detected anomalies varies significantly across layers (Figure 2). By combining the anomaly maps, a merged anomaly map representing different levels of complexity is generated. The sample-level prediction of anomalies is determined based on the pixel with the highest anomaly value within an instance, compared to a threshold. This method is preferred over averaging pixel anomaly scores to avoid disregarding small anomalies. At the pixel level, anomaly segmentation identifies all pixels exceeding a certain threshold as anomalous.

As detailed in Table 1, the teacher and student presentations are symmetrical in the corresponding layers. The teacher network employs downsampling with a kernel size of one and a stride of two, while the student network decodes with a kernel size of two and a stride of two. The OCBE module combines the three representations from the teacher network by adding two low-level

teacher representations and one mid-level teacher representation. This is followed by 3x3 convolutional layers, batch normalization, and a 1x1 convolutional layer with a stride of one.

Building upon an anomaly detection algorithm, the proposed method effectively detects diseased areas and calculates the leaf size-independent diseased leaf area percentage. Integrated into a single approach, these methods function in real-time without the need for preprocessing. Several modifications to the anomaly detection algorithm are implemented to enhance performance.

The proposed method integrates a masking algorithm for background segmentation, a disease detection model based on anomaly detection principles, and a calculation mechanism for determining the diseased leaf area percentage. Specifically, the RD++ model is utilized for anomaly detection with the following modifications:

- **Incorporation of a masking algorithm**
- **Modifications to the anomaly detection model**
 - **Alteration of output weights**

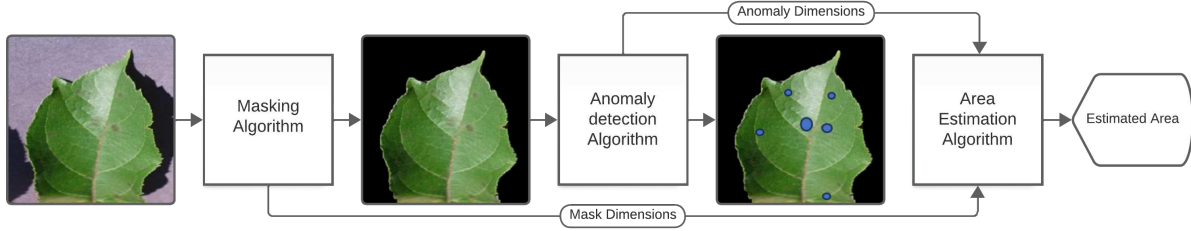


Figure 3: Flowchart of the Area estimation Process

- Addition of enhanced noise
- Adjustment of the colorspace
- Inclusion of diseased region calculation capability

Algorithm 1 Pseudo Code for Anomaly Detection

Require: Images

Ensure: maskedImage, diseasedArea

- 1: Load pre-trained anomaly detection model
 - 2: **for** image in images **do**
 - 3: maskedImage, initialMaskSize = LeafSelectionAlgorithm(image)
 - 4: anomalyMap = AnomalyDetectionModel(maskedImage)
 - 5: mask = anomalyMap $\geq$ predefinedThreshold
 - 6: secondaryMaskSize, anomalyMaskedImage = SecondaryMaskingAlgorithm(mask, maskedImage)
 - 7: diseasedArea = secondaryMaskSize / initialMaskSize
 - 8: **return** anomalyMaskedImage, diseasedArea
 - 9: **end for**=0
-

For a comprehensive understanding of the combined method, refer to the pseudo code in Algorithm 1 and the flowchart in Figure 3.

3.1 Leaf Selection Algorithm

There are several reasons for employing a mask around the leaf. Disease detection is enhanced when models are trained exclusively on the leaf area, excluding varying

backgrounds. The primary reason for using a mask is the ability to calculate the diseased area accurately. The diseased area of the leaf is determined by measuring the total leaf area. A mask defines the leaf area and size.

A leaf selection algorithm must meet several requirements. To maintain the project's unsupervised nature, without annotated data, the masking algorithm must also be unsupervised. Additionally, it must provide accurate segmentation, selecting the entire leaf area and constructing a mask around it. The algorithm should also utilize the mask size and dimensions to calculate the diseased area. Moreover, the chosen leaf selection algorithm should be relatively fast and reliable.

3.1.1 OpenCV

OpenCV [15] (Open Source Computer Vision Library) is a widely used open-source library for computer vision. Images are transformed into the LAB color space due to its resilience to changes in illumination and better sensitivity to green. An Otsu [18] and a binary inverted threshold are applied to the A channel, representing the green content in a pixel. Dilation and erosion are then performed to remove small noise and connect broken contours. Leaf contours are identified and filled with masks, which are merged with a black copy of the image, creating a black mask around the leaves.

3.1.2 Self-Supervised Segmentation

Inspired by the research of X. Lin *et al.* [14], a self-supervised approach was tested. This approach combines a self-supervised segmentation model with a color-based

leaf segmentation model. By adding superficial labels, traditional supervised machine learning techniques can be applied. Labels are generated by assigning the same label to embeddings that are close in the embedding space, as semantically similar pixels should be mapped close to each other. These artificial labels are used to train the model through backpropagation for each iteration until the maximum iteration is reached. A loss function penalizes different labels for adjacent pixels, reducing the number of labels. Utilizing the green characteristic of leaves, the most green labels are selected to create a binary mask containing only leaf objects.

3.2 Modifications to Detection Model

3.2.1 Weights

Visualizing different layers of anomaly maps revealed that low and mid-level anomaly maps provide better localization of diseased areas, while high-level anomaly maps are more general. When a , b , c , and d are the low, mid, high-level, and resulting anomaly maps respectively, the per-pixel anomaly score in RD++ is calculated as:

$$a + b + c = d, \quad (1)$$

treating all feature levels equally. Adding weights to the feature maps improved detection performance. The updated per-pixel anomaly score is calculated as follows:

$$7a + 12b + 4c = d. \quad (2)$$

3.2.2 Noise

To ensure the algorithm learns only the leaf features and not the background, the leaf contour is first calculated. Noise is then added, and the background is masked based on the contour. The Simplex-Noise used in the original paper [32] is small in size and intensity and does not resemble the encountered disease closely.

Using plant biology expertise, improved noise types were proposed to better simulate disease. Effective noise should match the observed disease color range, appear only in the leaf area, and be irregular in size. The original Simplex noise [32] creates small white noise areas, which are not representative of the encountered diseases. Two improved noise algorithms are proposed:

Name	Dimensions
Input- Image	$3 \times 256 \times 256$
Low-Level Teacher	$256 \times 64 \times 64$
Mid-Level Teacher	$512 \times 32 \times 32$
High-Level Teacher	$1024 \times 16 \times 16$
Low-Level Student	$256 \times 64 \times 64$
Mid-Level Student	$512 \times 32 \times 32$
High-Level Student	$1024 \times 16 \times 16$
Low-Level Compact Feature	$256 \times 64 \times 64$
Mid-Level Compact Feature	$512 \times 32 \times 32$
High-Level Compact Feature	$1024 \times 16 \times 16$
OCBE Feature	$2048 \times 8 \times 8$
Output Anomaly Map	$1 \times 256 \times 256$

Table 1: Overview of the Dimensions and Datatypes present in the different stages within the anomaly detection model.

OpenSimplex The first noise algorithm uses the **OpenSimplex** [28] algorithm, producing smoother noise with a larger kernel size but at a higher computational cost. For each image, a new seed generates different noise per image but consistent noise within the image. Four noise layers are created, each with a random scale and color from a defined range, resulting in organic noise with irregular shapes and fitting colors.

Random Points The second noise algorithm simulates round disease spots seen in the dataset. A random number of spots (between 1 and 40) are placed on the leaf, each with random coordinates and sizes. These spots are filled with a random color from a brownish color range and added to the leaf, replicating round disease spots.

The images used are initially coded in the RGB color space and subsequently converted to the BGR color space for model training. RGB-like color spaces are significantly correlated, meaning high pixel values in one color channel often correspond to high values in other channels [20]. Consequently, structural changes impact all RGB components.

Inspired by Schuler et al. [25], the CIE LAB color space was selected to enhance model performance and mitigate issues associated with the RGB color space. The LAB color space comprises three channels and is

Noise Type	Characteristics	Description
Simplex	Small, low intensity	Basic white noise, less representative of disease
OpenSimplex	Smooth, larger kernel	Organic noise, irregular shapes and fitting colors
Random Points	Random spots	Simulates round disease spots, varying sizes and colors

Table 2: Comparison of Simplex Noise and Improved Noise Algorithms. This table summarizes three noise types used for simulating diseased areas on leaves: Simplex (basic white noise), OpenSimplex (organic noise with fitting colors), and Random Points (mimics round disease spots).

an achromatic-chromatic space. Chromatic information (color) is represented separately from achromatic (brightness) information. The Luminance (L) channel represents lightness or gray levels, effectively separating lightness from color information. The A channel stores green to red hues, while the B channel stores blue to yellow hues. By decoupling image lightness from color information, the LAB color space allows for more illumination-independent analysis. Consequently, a model trained on the LAB color space should exhibit greater generalizability to images with varying lighting conditions and improved performance. Additionally, the A and B channels’ lack of lightness information enhances color contrast. The LAB color space’s channels exhibit low inter-channel correlation [20], making it well-suited for handling noise and structural variance [23].

To implement the color space transformation, images are converted to the LAB color space using OpenCV after masking and noise addition, and before normalization. During inference, the model is applied to the LAB images. Post-processing, including the calculation of disease coverage, is performed in the BGR color space to ensure uniformity in both training and inference processes.

3.3 Calculation of Diseased Area

The percentage of diseased leaf area is a valuable metric when related to the total leaf area, making it comparable across different leaves regardless of size or the distance between the camera and the leaf. By thresholding the combined anomaly map generated by the anomaly detection model, the diseased area can be calculated by summing the number of pixels above the threshold. This diseased area is then divided by the total leaf area, calculated in Section 3.1, and multiplied by one hundred to return the diseased area as a percentage of the leaf.

4 Results and Analysis

4.1 Implementation Details

Experimental Settings. Utilizing a WideResNet50 [33] as the backbone, as suggested in RD++, the models were trained for 400 epochs with a batch size of 2. The Adam Optimizer [12] was employed for optimization. The learning rate was set at 0.005 for both the student component and the OCBE module, while a lower rate of 0.001 was applied to the projection layers. The weighting of various losses was configured as follows: the distillation loss was assigned a weight of 1, and the weights for the SSOT loss, contrast loss, and reconstruction loss were set at 0.2, 0.02, and 0.002, respectively. This configuration, except for the batch size and epochs, mirrors that used in RD++.

4.1.1 Colorspace

Dataset. The dataset, available through the PlantVillage project [9], is an extensive collection of leaf images from various crops, categorized into healthy leaves and a variety of diseases. The dataset comprises over 54,000 images, divided into 14 crops and 38 different diseases. To demonstrate the applicability of the developed method, a subset of the PlantVillage dataset was selected. This subset included all images of *Malus domestica* (Apple), with 1,645 healthy leaf images and 630 images of Apple Scab. The healthy leaf images were split into 80% for training and 20% for testing. Testing was conducted on the images with Apple Scab (630 images) and the remaining healthy images. It is important to note that the distance from which the pictures were taken varies between the healthy and diseased leaf images. Most images of healthy leaves include the leaf stem, whereas the diseased images are closer and do not show the stem.

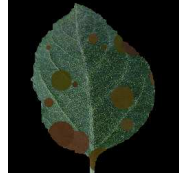
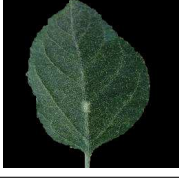
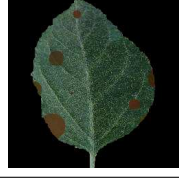
Simplex RD[29]	Opensimplex	Random Points
		
		

Table 3: Comparison of different noise added to a leaf image. In column one, the original noise proposed in [29] is seen. Small white areas are created here. In column two, noise is produces by Opensimplex. Organic shapes with colors close to the leaf and disease color are created. In column three, noise in form of points are used.

Evaluation Metrics. To evaluate the segmentation and classification for anomaly detection, the following metrics were used: AUROC image-wise, AUROC pixel-wise, PRO metric [3], F1 score, and Precision. These metrics assess the classification capabilities of the model. For image-wise classification, the highest anomaly score within the image is thresholded for a binary classification. For pixel-wise classification, each pixel is classified by thresholding the anomaly score for a binary decision.

The primary output of interest for this method is the area predictions, with segmentations serving as a means to obtain these predictions. Consequently, the exact location of the predicted leaf and disease segmentations is not critical; only their relative size matters. Therefore, the relevant metrics are not typical segmentation metrics that evaluate overlay with the ground truth, but metrics that assess the accuracy of the predicted sizes for the leaf, disease, and diseased leaf percentage relative to the ground truth. For the area calculations, which are continuous regression estimations, the metrics used include MAE, MSE, RMSE, MAPE, R-squared, and CCC.

These metrics collectively provide a robust framework for evaluating the performance of models in detecting and quantifying plant disease anomalies, ensuring that predic-

tions are accurate and reliable for practical applications.

4.2 Performance of Proposed Method

4.2.1 Leaf Selection Algorithm

Using the OpenCV library script provides a good segmentation, with a very fast inference time (Table 4). However it struggles with low contrasts in color, especially in outer regions where color changes due to diseases or shadow are present. Additionally, there’s a need to optimize the balance in the use of dilation and erosion techniques to adequately fill the diseased areas without excessively expanding beyond the leaf’s boundaries.

The Self supervised approach combines structural feature learning with color based thresholding, and is therefore more robust on outer regions of the leaf. A problematic weakness, is the cutting out of regions within the leaf, if they show both a strong contrast in structure and color due to disease. Due to the immense inference time, which was about 27 times longer than with the OpenCV method, the self supervised segmentation was not chosen. Proposed Method Instead, based on the faster inference time and good prediction results the OpenCV method was chosen. So, when referring to the leaf selection algorithm the OpenCV method is meant, as it is used in the final method. As shown in Table 5, when measured by the R-squared, the prediction results are quite good. Besides the good regression results, the segmentations are also good in the measured metrics. Some of the metrics used here, such as the MAE and MSE, are only relevant in comparison. Meaning, they can only state a performance difference when compared to a method used on the same data, since they are influenced by properties of the data with which they were calculated on. For better understanding the residuals area mapped. The leaf selection algorithm tends to underestimate the leaf area pretty constantly (Figure 4).

4.2.2 Anomaly Map Weights

By optimising the weights for the anomaly maps, which combine the final anomaly map, a strong improvement both the AUROC Pixel and PRO metrics are archived. Since the performance already was perfect on the classification task no performance improvements can be read

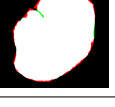
Image	Ground Truth	OpenCV	Self Supervised Segmentation
			
			
			
			

Table 4: Comparison of different unsupervised leaf segmentation algorithms. Color code: *Green* false negatives (leaf classified as background), *red* false positives (background wrongly classified as leaf), *black* true negatives (background detected as background), *white* true positive (leaf area classified as leaf). For a perfect segmentation only white and black should be present.

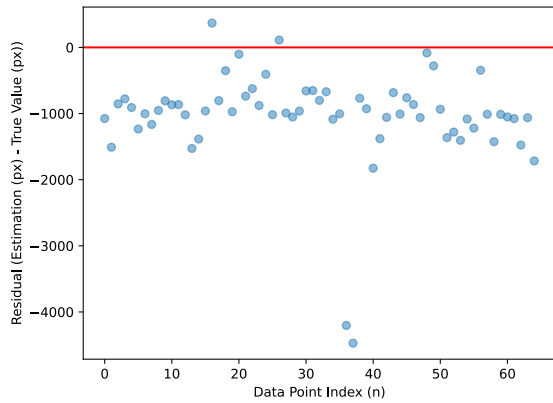


Figure 4: It presents the residual plots for leaf area estimation, showing the residuals (estimation error) for each data point (n). The red line indicates zero residual, where perfect estimation would lie.

from this, but the AUROC Pixel matrix and PRO matrix show improved segmentation results.

4.2.3 Noise

By comparing the original Simplex noise, used by RD++, the Point and Opensimplex noise, across a small annotated dataset and a big not annotated dataset, it was shown that the improved noises increased the model performance strongly. The annotated dataset gives insight on the segmentation and an indication for classification. The big dataset was used to validate the classification on a large number of images with large variation. The Opensimplex noise performed the best and had perfect classification results on the Big Dataset (Table 8).

4.2.4 Colorspace

The performance of the LAB colorspace was worse than the results by the RGB colorspace, for all performance metrics (Table 8). The performance was particular bad for the segmentation task. Due to the weak performance, the LAB colorspace model configuration was not tested on the Big Dataset or trained with the noise variations.

4.2.5 Proposed Method

Resulting is a disease detection model using the RGB colorspace with opensimplex noise. The model has problems detecting disease on the outer leaf parts and very small, disease spots. The latter fits with the low PRO metric result, with the high AUROC Pixel result (Table 8).

The disease area prediction have a high performance with a certain error, explain a good R-square, but a high MAPE, showing variation in both ways (Table 5). All healthy leaf images in the annotated dataset were detected correctly as healthy with no diseases detected.

4.2.6 Diseased Lea Area Percentage Calculation

The diseased leaf area percentage calculation is dependent on the leaf masking and the disease detection. The diseased leaf area percent predictions have a high R-squared combined with a relative high MAPE (Table 5). Again, as in the diseased area prediction, the healthy images are correctly detected as healthy also. The prediction results

	MAE	MSE	RMSE	MAPE	R-squared	CCC
Leaf Area	1046.34	1568087.97	1252.23	3.35	0.96	0.98
Diseased Area	812.6	2478619.52	1574.36	43.26	0.89	0.94
Diseased Leaf Area Percentage	2.45	23.11	4.81	45.17	0.84	0.93

Table 5: Evaluation metrics for the regression results on a annotated test set of 32 healthy and 32 diseased leaf images.

	Accuracy	IoU	Dice Coefficient
Leaf Area	0.98	0.95	0.98

Table 6: Evaluation metrics for the **segmentation** results for the leaf area on an annotated test set of 32 healthy and 32 diseased leaf images.

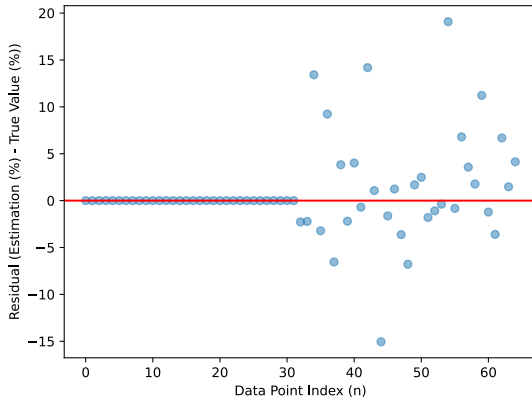


Figure 5: Residual plots for the diseased leaf area percentage. The 0 values at the data points are due to healthy images in the annotated inference dataset. The healthy leaves have no disease detected, and therefore the residuals are zero.

deviate as much as 20 percent from the ground truth values (Figure 5). To understand the combined results, the proposed method has to be evaluated separately. So, first, the leaf selection algorithm is evaluated, then the disease detection algorithm with its variations is compared before the leaf area estimation computation is evaluated.

5 Discussion

The performance of the unsupervised anomaly detection model was highly influenced by the dataset characteristics. The PlantVillage dataset, with its homogeneously strong green contrast between the background and the leaf, facilitated effective leaf segmentation using the OpenCV-based method. This approach yielded lower prediction errors in leaf area estimates by default, which could potentially be corrected by adjusting the image processing steps, such as increasing dilation or reducing erosion around leaf contours. Furthermore, the presence of outliers (Figure 4), necessitates a closer examination to determine if the discrepancies are due to inaccuracies in the ground truth masks or the prediction method itself.

Despite the successful application of the OpenCV method, which proved competitive with the supervised model developed by Shoaib et al. [27] by providing comparable accuracy, IoU, and Dice coefficients in a fraction of the time and without the need for training or annotations, there is still room for improvement. For instance, the self-supervised segmentation approach, though not utilized in the final model due to its high inference time, might still be enhanced with a smaller, optimized model.

In the realm of anomaly detection, direct comparisons with other methods were challenged by the limited application of our model to the Apple Scab subset of the PlantVillage dataset and the variety of evaluation metrics used in other studies, some of which do not report per-

	Weights							
Source	low level	mid level	high level	F1	Precision	AUROC Pixel	AUROC Image	PRO
RD++	1	1	1	1	1	0.904	1	0.55
Proposed	7	12	4	1	1	0.92	1	0.61

Table 7: Comparison of weights from RD++ and the proposed weights, done on a model trained with Opensimplex noise in the RGB colorspace.

		Annotated Dataset					Big Dataset	
Colorspace	Noise	F1	Precision	AUROC Pixel	AUROC Image	PRO	F1	Precision
RGB	Simplex	1	1	0.91	1	0.56	0.91	0.84
RGB	Point	1	1	0.92	1	0.60	0.99	0.98
RGB	Opensimplex	1	1	0.95	1	0.61	1	1
LAB	Simplex	0.88	0.78	0.36	0.54	0.07	-	-

Table 8: Inference for the compared methods. The annotated dataset consisted of 32 healthy and 32 annotated diseased leaf images with ground truth masks. The big dataset consisted of 200 healthy and 200 diseased leaf images without ground truth masks.

formance metrics [23, 31, 10]. Nonetheless, compared to the supervised methods [16], our method demonstrated improvements in precision and F1 scores, achieving perfect scores of 1.0. The analysis of model parameters like weights, noise, and colorspace revealed that fine-tuning specific to the dataset could enhance performance. Particularly, the use of OpenSimplex noise helped improve generalization capabilities, while switching from RGB to LAB colorspace did not yield beneficial results, possibly due to the pre-trained model’s backbone being optimized for RGB. Calculating the percentage of leaf area with disease emphasized how errors in predicting both the total leaf area and the diseased area can multiply each other. Our model’s ability to combine both predictions into a coherent estimation of diseased leaf area percentage, demonstrates its utility. The residuals from these calculations (Figure 5) suggest that while the model performs well within known data variance as indicated by solid R-squared values, the biggest improvements can be made by enhancing the accuracy of diseased area predictions.

The implementation of this unsupervised method offers a reliable alternative to manual estimations, providing consistent results with known precision and no drift over time, which marks a substantial improvement over

traditional methods. The method’s fast inference time and high accuracy make it suitable not only for applications in apple scab detection in apple cultivation [4] but also for other area estimation tasks where anomalies or diseases need to be quantified. Future applications could extend to diverse fields such as quantifying growth in petri dishes or counting flies on flypaper, underscoring the method’s versatility and potential for broad applicability in automated disease detection.

6 Conclusion

The improved unsupervised anomaly detection model, using a leaf segmentation algorithm based on OpenCV, demonstrated high accuracy and low error in predicting diseased leaf area percentages on the PlantVillage dataset. This method, which outperformed self-supervised segmentation methods, offers a state-of-the-art approach to unsupervised disease detection in plants, making it suitable for field use and reducing the need for manual labor. By analyzing model parameters, such as weight changes, noise, and color space, this approach provides valuable insights for future research and development in plant disease detection. This novel method simplifies the applica-

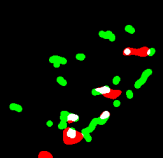
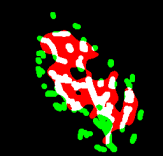
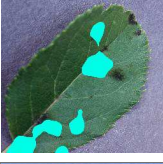
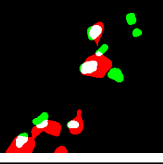
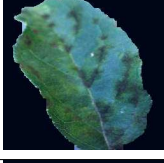
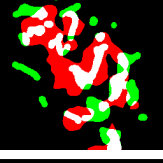
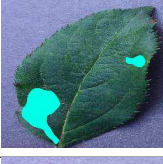
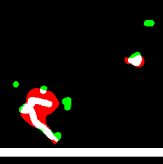
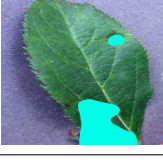
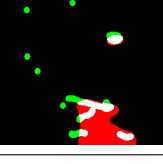
Masked Leaf	Detected Disease	Ground Truth Comparison
		
		
		
		
		
		

Table 9: Non-cherry-picked inference results showcasing the performance of the proposed method for disease detection on various leaf images. The middle column, highlighted in *turquoise*, indicates the areas where disease has been detected by the model. The color coding for the comparison with ground truth is as follows: *Green* represents false negatives (regions where the ground truth indicates disease, but the model did not detect it), *red* represents false positives (areas where the model incorrectly detected disease in the background), *black* represents true negatives (background correctly identified as disease-free), and *white* represents true positives (diseased areas correctly identified by the model).

tion of machine learning to plant breeding and disease detection by eliminating the need for disease-specific training data, thus facilitating broader and more efficient deployment, particularly in real-time applications. Further testing across different species, diseases, and tasks is suggested to fully explore its potential.

References

- [1] K. Batzner, L. Heckler, and R. König. Efficientad: Accurate visual anomaly detection at millisecond-level latencies, 2023. [2](#)
- [2] A. Benfenati, P. Causin, R. Oberti, and G. Stefanello. Unsupervised deep learning techniques for automatic detection of plant diseases: reducing the need of manual labelling of plant images. *Journal of Mathematics in Industry*, 13, 05 2023. [2](#)
- [3] P. Bergmann, M. Fauser, D. Sattlegger, and C. Steger. Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2020. [8](#)
- [4] D. Biddinger. *Compendium of Apple and Pear Diseases and Pests 2nd edition*, pages 161–166. 01 2014. [11](#)
- [5] C. H. Bock, P. E. Parker, A. Z. Cook, and T. R. Gottwald. Visual rating and the use of image analysis for assessing different symptoms of citrus canker on grapefruit leaves. *Plant Disease*, 92(4):530–541, 2008. PMID: 30769647. [1](#)
- [6] J. C. and M. S. A comprehensive review on detection of plant disease using machine learning and deep learning approaches. *Measurement: Sensors*, 24:100441, 2022. [2](#)
- [7] A. S. Capucho, L. Zambolim, H. S. S. Duarte, and G. R. O. Vaz. Development and validation of a standard area diagram set to estimate severity of leaf rust in *coffea arabica* and *c. canephora*. *Plant Pathology*, 60(6):1144–1150, 2011. [1](#)
- [8] H. Deng and X. Li. Anomaly detection via reverse distillation from one-class embedding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9737–9746, June 2022. [2](#)
- [9] D. P. Hughes and M. Salathé. An open access repository of images on plant health to enable the development of mobile disease diagnostics through machine learning and crowdsourcing. *CoRR*, abs/1511.08060, 2015. [7](#)
- [10] R. Katafuchi and T. Tokunaga. Image-based plant disease diagnosis with unsupervised anomaly detection based on reconstructability of colors. *CoRR*, abs/2011.14306, 2020. [2](#), [11](#)

- [11] K. KC, Z. Yin, M. Wu, and Z. Wu. Depthwise separable convolution architectures for plant disease classification. *Computers and Electronics in Agriculture*, 165:104948, 2019. 2
- [12] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014. 7
- [13] R. Likert. A technique for the measurement of attitudes. *Archives of Psychology*, 140:1–55, 1932. 1
- [14] X. Lin, C.-T. Li, S. Adams, A. Kouzani, et al. Self-supervised leaf segmentation under complex lighting conditions. *Pattern Recognition*, page 109021, 2022. 5
- [15] N. Mahamkali and V. Ayyasamy. Opencv for computer vision applications. 03 2015. 5
- [16] O. Mzoughi and I. Yahiaoui. Deep learning-based segmentation for disease identification. *Ecological Informatics*, 75:102000, 2023. 2, 11
- [17] F. Nutter, Jr and P. Schultz. Improving the accuracy and precision of disease assessments: Selection of methods and use of computer-aided training programs. *Canadian Journal of Plant Pathology*, June 1995:174–184, 12 2009. 1
- [18] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, 1979. 5
- [19] D. S. Paul Bergmann, Michael Fauser and C. Steger. Mvtec ad—a comprehensive real-world dataset for unsupervised anomaly detection. In *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition*, page 9592–9600, 2019. 2
- [20] T. Pouli, E. Reinhard, and D. W. Cunningham. *Image Statistics in Visual Computing*. A. K. Peters, Ltd., USA, 1st edition, 2013. 6, 7
- [21] K. Roth, L. Pemula, J. Zepeda, B. Schölkopf, T. Brox, and P. V. Gehler. Towards total recall in industrial anomaly detection. *CoRR*, abs/2106.08265, 2021. 2
- [22] G. Russell. *Plant Breeding for Pest and Disease Resistance: Studies in the Agricultural and Food Sciences*. Elsevier Science, 2013. 1, 2
- [23] J. P. Schwarz Schuler, S. Romani, M. Abdel-nasser, H. Rashwan, and D. Puig. Reliable deep learning plant leaf disease classification based on light-chroma separated branches. 10 2021. 7, 11
- [24] J. P. Schwarz Schuler, S. Romani, M. Abdel-nasser, H. Rashwan, and D. Puig. Color-aware two-branch dcnn for efficient plant disease classification. *Mendel*, 28:55–62, 06 2022. 2
- [25] J. P. Schwarz Schuler, S. Romani, M. Abdel-nasser, H. Rashwan, and D. Puig. Color-aware two-branch dcnn for efficient plant disease classification. *Mendel*, 28:55–62, 06 2022. 6
- [26] N. K. Shivdutt Dixit. A review of approaches to classify plant disease using deep learning and machine learning. *2022 5th International Conference on Contemporary Computing and Informatics (IC3I)*, 2022. 2
- [27] M. Shoaib, T. Hussain, B. Shah, I. Ullah, S. M. Shah, F. Ali, and S. H. Park. Deep learning-based segmentation and classification of leaf images for detection of tomato plant disease. *Frontiers in Plant Science*, 13, 2022. 10
- [28] K. Spencer and O. Raccuglia. Opensimplex noise. <https://github.com/lmas/opensimplex>. 6
- [29] T. D. Tien, A. T. Nguyen, N. H. Tran, T. D. Huy, S. T. Duong, C. D. T. Nguyen, and S. Q. H. Truong. Revisiting reverse distillation for anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24511–24520, June 2023. 2, 8
- [30] X. Wang, Z. Yu, S. D. Mello, J. Kautz, A. Anandkumar, C. Shen, and J. M. Alvarez. Freesolo: Learning to segment objects without annotations, 2022. 2
- [31] J. Wu, C. Wen, H. Chen, Z. Ma, T. Zhang, H. Su, and C. Yang. Ds-detr: A model for tomato leaf disease segmentation and damage evaluation. *Agronomy*, 12(9), 2022. 2, 11
- [32] J. Wyatt, A. Leach, S. M. Schmon, and C. G. Willcocks. Anoddpm: Anomaly detection with denoising diffusion probabilistic models using simplex noise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 650–656, June 2022. 3, 6
- [33] S. Zagoruyko and N. Komodakis. Wide residual networks. In R. C. Wilson, E. R. Hancock, and W. A. P. Smith, editors, *Proceedings of the British Machine Vision Conference 2016 (BMVC 2016)*. BMVA Press, 2016. 7
- [34] S. Zhang and C. Zhang. Modified u-net for plant diseased leaf image segmentation. *Computers and Electronics in Agriculture*, 204:107511, 2023. 2
- [35] Q. Zou, Y. Yang, W. Y. Cheung, C. Liu, and X. Ji. Ilsgan: Independent layer synthesis for unsupervised foreground-background segmentation, 2023. 2