NHL STENDEN
computer vision & data science

Visual Prompt and Reference-Guided Plant Health Detection and Classification for Precision Agriculture

NHL Stenden Professorship in Computer Vision & Data Science

Athul Dev,
Supervisor: Willem Dijkstra

Abstract—Ensuring potato plant health is critical to mitigating crop losses caused by diseases such as Potato Virus Y (PVY). Traditional manual visual inspections are labor-intensive and often inaccurate, emphasizing the need for automated solutions. This research explores the application of RGB image analysis to segment and classify potato leaves, leveraging state-of-the-art deep learning models to address the challenges posed by agricultural datasets with diverse and complex imagery.

The images were taken using a camera mounted on a moving cart, positioned to face the ground, which introduced challenges such as blurry images due to movement. The study involved systematically filtering blurry images, leveraging user defined points and boxes as prompts, along with exploring the possibility of reference image as a guide for classification and evaluating instance segmentation techniques. Models such as ViT, ResNet, and EfficientNet were trained for classification, while segmentation relied on SAM2 without the need for fine-tuning.

Segmentation tasks were performed to extract individual leaf data from the images, which was then fed to the classifier to determine whether the leaves were infected with PVY. This approach highlights the crucial interplay between segmentation and classification tasks in addressing the problem. ViT exhibited exceptional robustness to image quality variations, while embedding-based classification, particularly majority voting, enabled flexible, plug-and-play modularity. In terms of prompt heuristics, Point-based prompting achieved the highest IoU scores for segmentation, with box-based and edge-based prompting offering strengths in specific contexts.

Results highlight that dataset diversity and size are more influential than preprocessing techniques like removing out-of-focus images. Moreover, by decoupling segmentation and classification, the approach enables easy integration of new classification models, making it adaptable to diverse and broader contexts. Additionally, by leveraging SAM2, enabled the generation of instance-level segmentation masks and bounding boxes facilitating quicker annotations, enabling subject matter experts to efficiently review and refine datasets.

Index Terms—RGB Image Analysis, SAM2, ViT, ResNet, EfficientNet, Embedding Comparison, Instance Segmentation, PVY.

1 Introduction

The automation of plant variety identification and disease diagnosis in agriculture has become essential as traditional methods grow less viable. Manual inspections, heavily reliant on the expertise of farmers and agronomists, are both time-intensive and susceptible to human error [1, 2, 3, 4, 5, 6, 7]. In regions facing labor shortages, such as the Netherlands, this issue is further amplified. Advanced computer vision techniques can address these challenges by offering quicker, more consistent plant classification and disease detection solutions [8, 9, 10].

Advances in deep learning and computer vision have greatly improved the detection of early and late blight in potato plants [11, 12, 13, 14]. However, Potato Virus Y (PVY) remains a significant economic threat due to the complexity of its symptoms, which are subtle and difficult to classify [15]. According to the U.S. Department of Agriculture (USDA), PVY often manifests as a mosaic pattern on the leaves, typically appearing as green and yellow patches, though it can also present as leaf wrinkling (Figures 1b and 1c) [16].

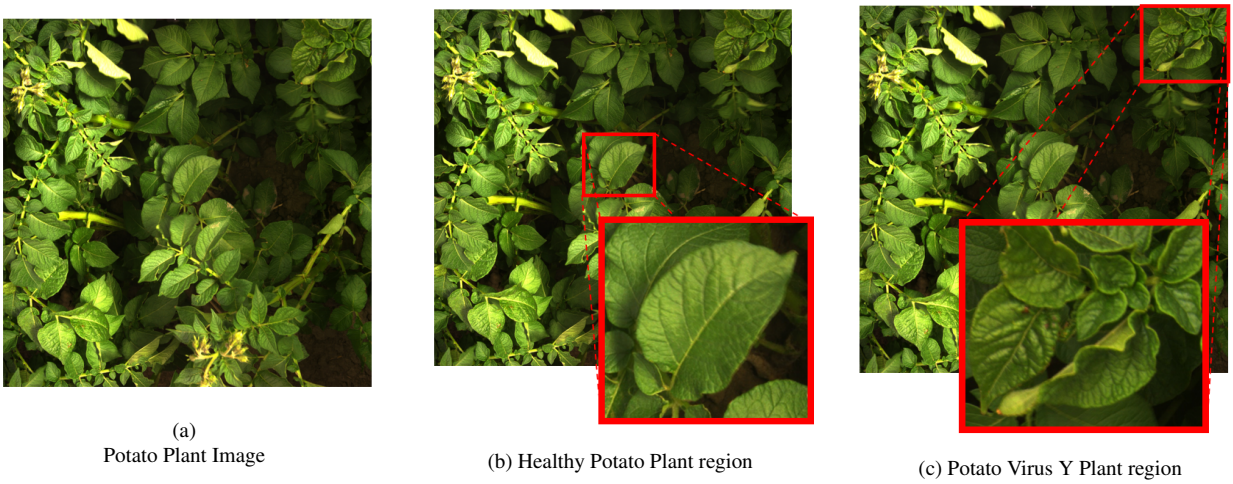
Existing methods for identifying PVY-infected plants heavily rely on hyperspectral imaging [17, 18, 19]. While hyperspectral data can provide detailed insights, the high cost of such cameras makes this approach impractical for widespread use. Additionally, the symptoms of PVY can vary significantly depending on the potato cultivar, as different varieties respond uniquely to the infection [20]. These differences complicate the development of generalized models capable of accurately identifying PVY across diverse potato populations, highlighting the need for cost-effective and adaptable solutions.

In response to this challenge, this research focuses on a two-stage mechanism to develop a decoupled approach that separates segmentation and classification to effectively address these gaps. The model's objective is to navigate the complexities of dense and intricate images, identifying and classifying regions of interest to evaluate the overall health of potato plants. By leveraging user-provided visual prompts, such as points and boxes, for segmentation and incorporating an embedding-based search method using reference images for classification, the segmentation stage identifies regions of interest, while the classification stage determines whether these regions are healthy or PVY-affected. The approach not only aids in PVY detection but also enhances the model's practical usability, enabling agricultural experts to perform quicker and more accurate annotations while adapting seamlessly to diverse and demanding agricultural settings.

To guide this exploration, the study evaluates whether a prompt-based model can effectively segment individual leaves and, in combination with a fine-tuned classifier, accurately classify these regions as healthy or PVY-affected in dense images, where traditional methods often falter. At the heart of this inquiry lies the goal of determining how effectively such a model can generalize across diverse scenarios, such as making it applicable to multiple potato cultivars by allowing users to provide a reference image to identify and group similar leaves. Another scenario involves adapting to changes in leaf characteristics caused by evolution or seasonal variations, ensuring consistent performance. Together, these scenarios enhance the model's adaptability and streamline annotation processes for agricultural experts.

This research is structured to explore the potential of these

- *Athul Dev is a Computing Science student at the NHL Stenden University of Applied Sciences, E-mail: athul.athul.dev@student.nhlstenden.com*
- *Willem Dijkstra is a researcher at the NHL Stenden Professorship in Computer Vision & Data Science, E-mail: willem.dijkstra@nhlstenden.com.*



(a)
Potato Plant Image

(b) Healthy Potato Plant region

(c) Potato Virus Y Plant region

Fig. 1: **Potato Plant Foliage**: (a) An example of a potato plant from the dataset, taken using the camera mounted on the moving cart. (b) An example of a healthy potato plant highlighting the foliage's texture. (c) An example of a Potato Virus Y diseased potato plant highlighting the mosaic pattern on the leaves.

methods for accurately detecting and classifying PVY, with a focus on prioritizing recall. By minimizing false negatives, the approach ensures that PVY infections are promptly identified, preventing their unnoticed spread and reducing the risk of significant crop losses.

2 State of the Art

Detecting PVY-infected leaves in dense, complex environments poses unique challenges. Occlusions, diverse backgrounds, and high multi-class variability, which arise from healthy and infected leaves at different infection stages, along with extraneous elements such as dirt, plant stems, and foreign objects, complicate accurate identification. These factors introduce overlapping visual features, inter-class similarities, and intra-class variability, making the detection process particularly challenging. Traditional category-specific object detection and segmentation models [21] rely on pre-defined categories, assuming that the test set closely resembles the training set. These models struggle to generalize in such heterogeneous environments, where objects exhibit inter-class similarities, intra-class variability, and overlapping structures. Transformer-based models [22] [23], such as Deformable DETR [24], introduced attention mechanisms to process multi-scale inputs, improving detection accuracy and efficiency. While these innovations perform well in structured environments, they face limitations in dense and variable scenes with occlusions and high object complexity. Addressing these limitations, advancements in object detection and segmentation now leverage user prompts for enhanced adaptability, enabling more precise and robust detection in challenging scenarios like PVY-infected leaf identification.

Instance segmentation, which assigns pixel-level labels to objects, has gained traction as a means of overcoming these limitations. Prompt-based segmentation models, such as SAM (Segment Anything Model) [25] and its derivatives like SAM2 [26], represent a paradigm shift by incorporating user-defined visual cues. These prompts, including bounding boxes, points, or textual descriptions, guide the model in identifying and segmenting specific regions of interest within an image. By enabling user-guided segmentation, SAM/SAM2 provides adaptability across diverse applications, including precision agriculture, where annotated datasets are scarce, and scenes are characterized by dense foliage, overlapping objects, and occlusions.

SAM2, at its foundation, the image encoder processes images into tokens that capture distinct features. These tokens act as the building blocks for all subsequent operations. SAM2 utilizes an MAE (Masked Autoencoder) pre-trained Hiera encoder, which excels at recognizing patterns even when parts of the data are missing. Its self-attention mechanisms enable SAM2 to analyze images in detail, capturing fine-grained features critical for accurate segmentation.

User interaction is seamlessly integrated through SAM2's prompt encoder and mask decoder. These components interpret inputs like clicks, scribbles, or bounding boxes to generate precise segmentation masks for objects of interest. The model's "two-way" transformer blocks dynamically update prompts and image embeddings, enabling SAM2 to refine predictions even when inputs are ambiguous. This adaptability allows SAM2 to cater to diverse user requirements, such as accurately detecting and segmenting PVY-infected regions on potato leaves under challenging conditions like partial occlusions or low contrast, while maintaining high accuracy.

To efficiently handle complex scenes, SAM2 employs a memory encoder to condense segmentation outputs into summaries. These summaries are stored in a memory bank, which uses a FIFO system to manage information and retain only the most relevant data. Additionally, SAM2's memory attention mechanism enables it to incorporate contextual information from earlier steps in the segmentation process. This enhances its ability to segment objects even in challenging scenarios involving occlusions or overlapping objects.

These capabilities are underpinned by SAM2's extensive training on the SA-V dataset, which includes over 600,000 spatio-temporal masks (masklets) and the SA-1B image dataset, providing SAM2 with unparalleled generalization capabilities. While SAM2 was originally designed with video segmentation in mind, its training also emphasizes image segmentation, ensuring high performance across diverse domains. The dataset's geographic diversity and comprehensive object coverage allow SAM2 to excel even in complex image datasets.

The architectural flow of SAM2 as shown in the Figure 2,builds on these foundational capabilities, processing images step-by-step to achieve seamless segmentation. The image encoder extracts features, acting as the model's foundation by identifying critical details in each image. Memory attention ensures consistent and coherent segmentation, even in scenarios where objects, such as leaves, exhibit subtle variations in appearance.

User prompts, such as clicks or bounding boxes, are fed into the prompt encoder to guide the segmentation process. These prompts help SAM2 focus on specific objects of interest, which are then segmented with precision by the mask decoder. The mask decoder combines features from the image encoder and guidance from the prompt encoder to produce highly accurate segmentation masks.

An alternative unified approach is represented by the SEEM

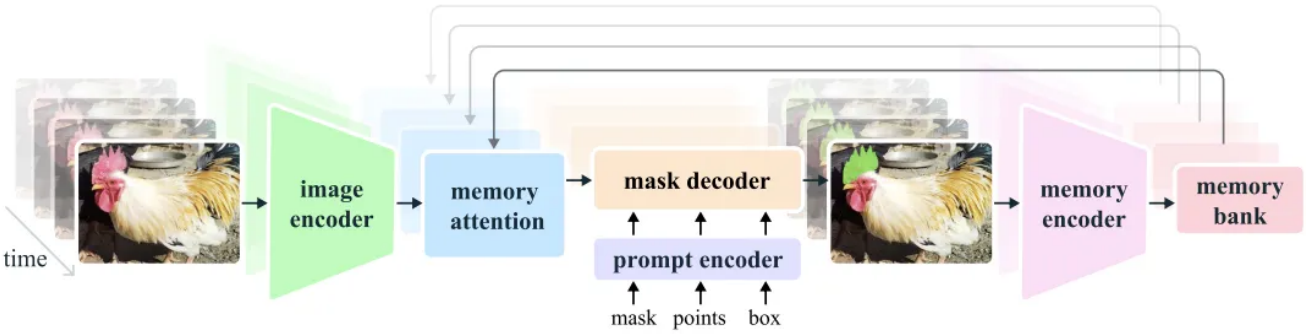


Fig. 2: **SAM2 Architecture**

(Segment Everything Everywhere All at Once) model[27], which integrates segmentation, detection, and classification into a single workflow. While SEEM simplifies the pipeline by requiring segmentation masks as input data, its dependence on pixel-level annotations for training and fine-tuning can be a limitation in resource-constrained domains like agriculture. Additionally, the all-in-one nature of SEEM may restrict the flexibility to customize or optimize individual components, making it less adaptable to highly variable conditions.

In contrast, SAM2 provides a focused and flexible solution for segmentation by leveraging user-defined prompts to isolate regions of interest. However, SAM is inherently limited to segmentation and does not perform classification. To address this gap, a two-stage pipeline is proposed, where segmentation is performed first using SAM2, followed by a separate classification model, as outlined in the Section 3.5.

Parallel to advancements in segmentation, vision transformers (ViTs) [28] have revolutionized the field of image classification. Unlike convolutional neural networks (CNNs) [29], ViTs employ self-attention mechanisms to model global relationships across the image, enabling them to capture fine-grained details and subtle patterns. This capability is particularly advantageous for classifying objects with nuanced differences, such as distinguishing between healthy and PVY leaves based on texture, mosaic patterns, color variations, or disease spots [30]. ViTs have demonstrated state-of-the-art performance across numerous image classification benchmarks, establishing themselves as a cornerstone of modern deep learning pipelines.

The second stage of the proposed pipeline employs vision transformers (ViTs) or convolutional neural networks (CNNs) for analyzing the segmented regions. ViTs, with their self-attention mechanisms, excel at capturing fine-grained details and subtle patterns, making them ideal for classifying objects with nuanced differences, such as distinguishing between healthy and diseased potato leaves. On the other hand, CNNs are well-suited for scenarios requiring efficient and scalable solutions.

This two-stage design offers significant advantages over unified models like SEEM. By decoupling segmentation and classification, each stage can be independently optimized, for instance, prompts can be fine-tuned for SAM2, while the classifier model can be trained or adjusted separately, ensuring robust performance across diverse use cases. The modularity of this pipeline supports a plug-and-play approach, enabling seamless integration of different segmentation models or classifiers tailored to specific user needs or domain requirements.

3 Materials and Methods

This section will discuss the materials and methods that were used throughout the research. This includes the dataset, segmentation and classification models, evaluation metrics, and prompting techniques.

3.1 Data Collection

The dataset used in this study consisted of high-resolution images of potato plants from a private collection. The images were taken using a camera mounted on a moving cart, positioned to face the ground as shown in Figure 1a. A total of 32 images, each measuring 5120x5120 pixels, were collected. Domain experts annotated these images in YOLO format, providing bounding boxes and labels for most of the leaves in the images. Each image, averaging 72 annotations, classified the leaves as either healthy or unhealthy (PVY) based on their visual characteristics. The distribution of annotations was uneven, with some images containing more PVY annotations than others.

3.2 Data Processing

The dataset preparation for classifying healthy and PVY-infected potato leaves followed a well-defined pipeline that emphasized quality and consistency. The process involved multiple interconnected steps, each tailored to improve the dataset's suitability for training, validation, and testing the classifier models.

3.2.1 Dataset Splitting and Preprocessing

The dataset comprised high-resolution images (5120x5120) annotated in the YOLO format, which were divided into training, validation, and test splits. To prevent any overlap of leaves of the same plant ending up in different datasets, that is between the training, validation or the test set, samples were grouped by filenames, and a stratified splitting technique was employed. This ensured that the distribution of "Healthy" and "PVY-Infected" classes remained consistent across all splits. By eliminating data leakage, this step ensured that no bias was introduced into the model, allowing for an accurate assessment of its performance.

3.2.2 Data Quality Improvement via Blur Detection

Another aspect of this pipeline was the addition of a blur detection mechanism to evaluate how the model's performance varied based on the inclusion or exclusion of blurry images. This step introduced a user-configurable element, allowing the dataset to be dynamically filtered based on sharpness thresholds provided at training time. Using the variance of the Laplacian operator, each image was quantitatively assessed for sharpness, and thresholds were calculated to classify images as "blurry".

The Laplacian operator is a second-order derivative operator that measures the rate of change in intensity across pixels in an image, highlighting edges and texture. Mathematically, for an image $I(x,y)$, the Laplacian is defined as:

$$L(x,y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2} \tag{1}$$

In practice, this operation is implemented using a convolution kernel, such as:

$$I = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \quad (2)$$

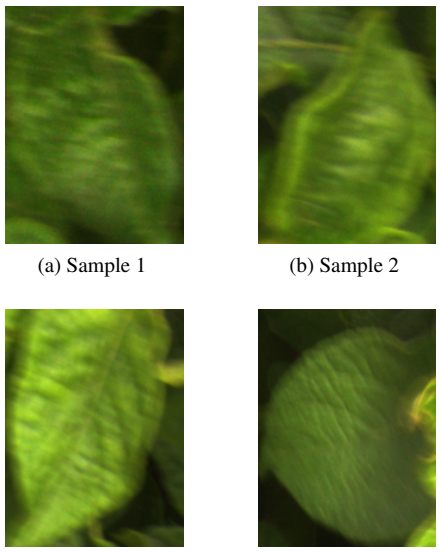
The output of the Laplacian operation emphasizes regions of high spatial frequency, such as edges, where intensity changes rapidly. To evaluate sharpness, the variance of the Laplacian is computed across the image:

$$\text{Variance} = \frac{1}{N}\sum_{i=1}^{N}(L_i - \mu)^2 \quad (3)$$

where L_i is the Laplacian value of the i-th pixel, μ is the mean Laplacian value, and N is the total number of pixels. A low variance value indicates a lack of high-frequency details (blur), whereas a high variance value corresponds to sharper images.

During data preprocessing, the blur threshold determined based in the statistics from the Fig 4 was used to classify images. Images with a variance below the threshold were marked as "blurry," while those above the threshold were considered normal. This threshold was configurable using the standard deviation offset value that could be defined by the user, enabling users to tailor the dataset for specific experimental conditions.

During training, we evaluated the impact of including or excluding blurry samples 3 on the classifier's performance. By excluding blurry images, the model trained on sharp, well-defined samples, enabling us to investigate whether this preprocessing step improved classification performance, particularly in dense and complex images. Conversely, retaining blurry images allowed us to assess the model's robustness and adaptability to varying data quality. This approach provided critical insights into whether training exclusively on sharp images or incorporating a threshold of blurry samples could enhance the model's overall effectiveness.



(a) Sample 1 (b) Sample 2

(c) Sample 3 (d) Sample 4

Fig. 3: **Sample of blurry images in dataset**: Illustrative examples of blurry images from the dataset

By incorporating the Laplacian-based blur detection mechanism, the pipeline enabled precise control over data quality, contributing to a deeper understanding of its impact on model performance.

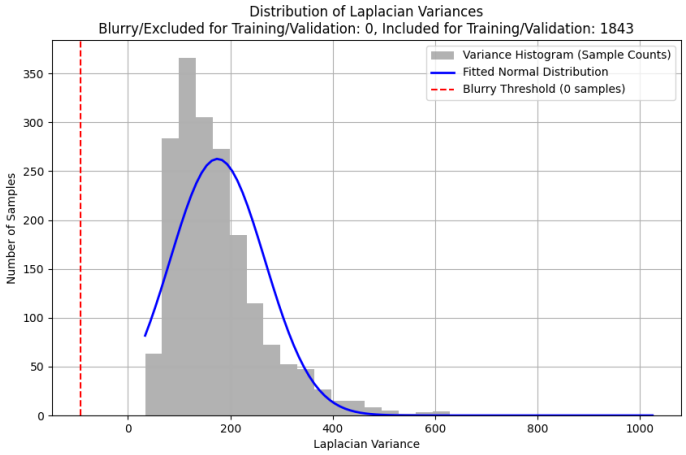
3.2.3 Stratified Cross-Validation

To ensure robust evaluation, the original high-resolution images were first divided into subsets, and individual leaves were extracted for further processing. The training and validation sets were partitioned into k-folds using stratified cross-validation, ensuring class balance at the per-leaf image level across all folds, as shown in Table 1. This approach allowed the model to be trained and validated on different subsets of the data in each iteration, ensuring comprehensive evaluation. Cross-validation provided valuable insights into the model's generalizability while reducing the risk of overfitting.

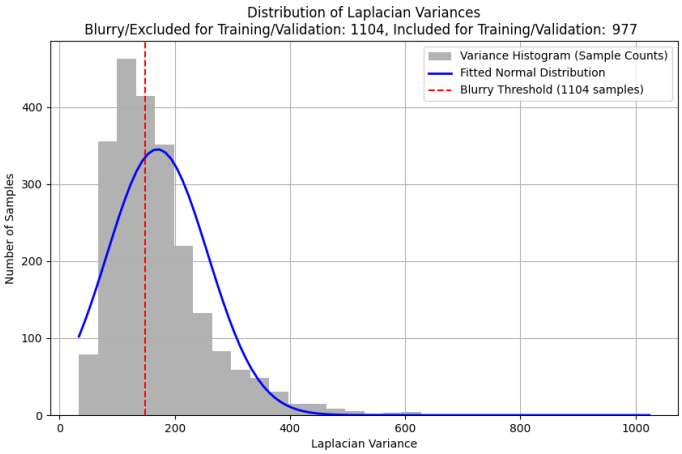
Table 1: Overview of healthy and PVY-infected percentages for train, validation, and test datasets.

Average Per-leaf Dataset Statistics Across k = 3 Folds

Dataset	**Total Samples**	**Healthy Samples**	**PVY Samples**
Train	1229	652 (53.1%)	577 (46.9%)
Validation	614	326 (53.1%)	288 (46.9%)
Test	448	198 (44.2%)	250 (55.8%)
Total	2291	1176 (51.3%)	1115 (48.7%)



(a) Baseline



(b) After Blur Removal

Fig. 4: **Visualization of Data Distribution During Data Loading in the Classifier Model Training Phase**: The plot shows the normal distribution of Laplacian variances. The 'Blurry/Excluded for Training/Validation' category represents out-of-focus samples with variances below the threshold, while the 'Included for Training/Validation' category comprises samples with variances above the threshold.

3.2.4 Visualization and Statistical Analysis

Visualization was employed to validate the effectiveness of preprocessing and filtering. The distribution of Laplacian variance values was analyzed for the train and validation datasets, containing per-leaf samples, to ensure that the removal of blurry or out-of-focus samples was processed correctly. The Figure 4 (a) represents the baseline distribution, which includes all samples and corresponds to a threshold of -2.0 standard deviations from the mean. In contrast, the Figure 4 (b) illustrates the filtered dataset with a threshold of -0.25 standard deviations, showcasing the effect of applying the blur detection thresholds. This analysis confirmed the successful exclusion of low-quality samples while retaining sufficient data for reliable training and evaluation. Additionally, the test set was excluded from this preprocessing to ensure it remained unbiased.

To ensure a robust evaluation, performance tracking across varying levels of data quality, such as blur scores in terms of Laplacian variance, was conducted to identify scenarios where the model might underperform. This analysis underscores the impact of image sharpness on model performance, providing valuable insights into its limitations and strengths. Detailed metrics and performance trends across different blur levels are discussed extensively in Section 4.1.

3.2.5 Data Augmentation

In this study, we did not apply any data augmentation techniques such as flipping, rotation, or synthetic noise generation. This decision was made to isolate the effects of the dataset's natural variability, enabling a clearer analysis of the models' robustness to variations in image quality and content. By focusing solely on the original data, we aimed to evaluate the inherent performance of classifiers under real-world conditions.

3.3 Segmentation Models

SAM2's robust architecture and working as explained in the Chapter 2 makes it an exceptional tool for image segmentation tasks, combining innovative components to achieve precise and efficient results.

These capabilities make SAM2 particularly effective for dense image segmentation, where it excels in isolating multiple objects with overlapping boundaries. Its ability to generate high-quality segmentation masks with minimal user input highlights its efficiency and adaptability. By utilizing this robust design, SAM2 efficiently processes high-resolution images and maintains consistency, even in complex scenes.



Fig. 5: **SAM2 Single Pass Mask Generation**

This is why we chose SAM2 for this study. Its prompt-based segmentation approach and ability to segment objects without fine-tuning make it a lightweight and highly versatile solution. SAM2's automatic mask generation feature, works by sampling single-point input prompts in a grid over the image, from each of which SAM can predict multiple masks. Then, masks are filtered for quality and deduplicated using non-maximal suppression. Additional options allow for further improvement of mask quality and quantity, such as running prediction on multiple crops of the image or postprocessing masks to remove small disconnected regions and holes. This produces segments of most leaves in a single pass as seen in the Figure 5, was especially beneficial. This enabled rapid data extraction for classifier pipelines and accelerated annotation workflows by providing bounding boxes for unannotated leaves.

For this study, SAM2 served as the primary segmentation model to delineate regions of interest (ROIs) from high-resolution potato leaf images. These segmented regions were subsequently classified using downstream models, such as Vision Transformers, to identify categories like "Healthy" or "PVY." SAM2's role as the segmentation front-end in this two-stage pipeline underscores its ability to enhance both data preparation and analysis. By leveraging SAM2, this study demonstrates how prompt-based, zero-shot segmentation models can streamline workflows and deliver high performance in specialized domains such as plant health monitoring.

3.4 Classification Models

As described in the state-of-the-art chapter 2, a variety of advanced architectures have been employed for plant disease classification. In this study, we explored Vision Transformers (ViT), ResNet, and EfficientNet architectures for classifying healthy and PVY-infected potato leaves. The base implementations of these models were sourced from the torchvision library, ensuring standardized and reliable performance.

ViT, as a relatively newer paradigm in computer vision, has shown exceptional performance in tasks requiring fine-grained feature extraction and global context understanding [31]. On the other hand, CNN-based architectures like ResNet and EfficientNet continue to provide robust baselines for image classification due to their architectural simplicity and computational efficiency. By leveraging the pretrained weights and configurations from torchvision, we ensured a strong foundation for benchmarking these architectures in the context of plant disease classification.

3.4.1 Vision Transformers (ViT)

Vision Transformers (ViT) represent a shift from convolutional neural networks to transformer-based architectures in computer vision. Unlike CNNs, which rely on localized feature extraction through convolutional layers, ViT divides images into fixed-size patches and processes them as sequences. Each patch is linearly projected into a token, and these tokens are passed through multiple transformer layers. This approach enables the model to capture global dependencies in the image, which is particularly useful for dense datasets like the one used in this study.

3.4.2 ResNet

ResNet (Residual Networks) [32] introduced a transformative approach to deep learning by incorporating residual connections, or skip connections, into convolutional neural networks. Unlike traditional architectures that directly stack layers, ResNet uses these connections to allow the flow of identity mappings, mitigating challenges like the vanishing gradient problem [33] and enabling efficient training of deeper networks.

3.4.3 EfficientNet

EfficientNet [34] represents an interesting approach to deep learning model design by leveraging a compound scaling method that simultaneously optimizes network depth, width, and resolution. Unlike traditional architectures that scale these dimensions independently, EfficientNet employs a systematic strategy to balance them, resulting in models that are both highly efficient and accurate. This approach allows EfficientNet to achieve state-of-the-art performance with fewer parameters and reduced computational requirements.

In this study, three models were fine-tuned for the binary classification of potato leaves into "Healthy" and "PVY" categories: Vision Transformer (ViT-B/16), ResNet18, and EfficientNet-B0. The ViT-B/16 model, with a patch size of 16x16 pixels, leverages its transformer-based architecture and self-attention mechanism to effectively process high-resolution images and capture subtle variations in leaf textures indicative of health status. ResNet18,

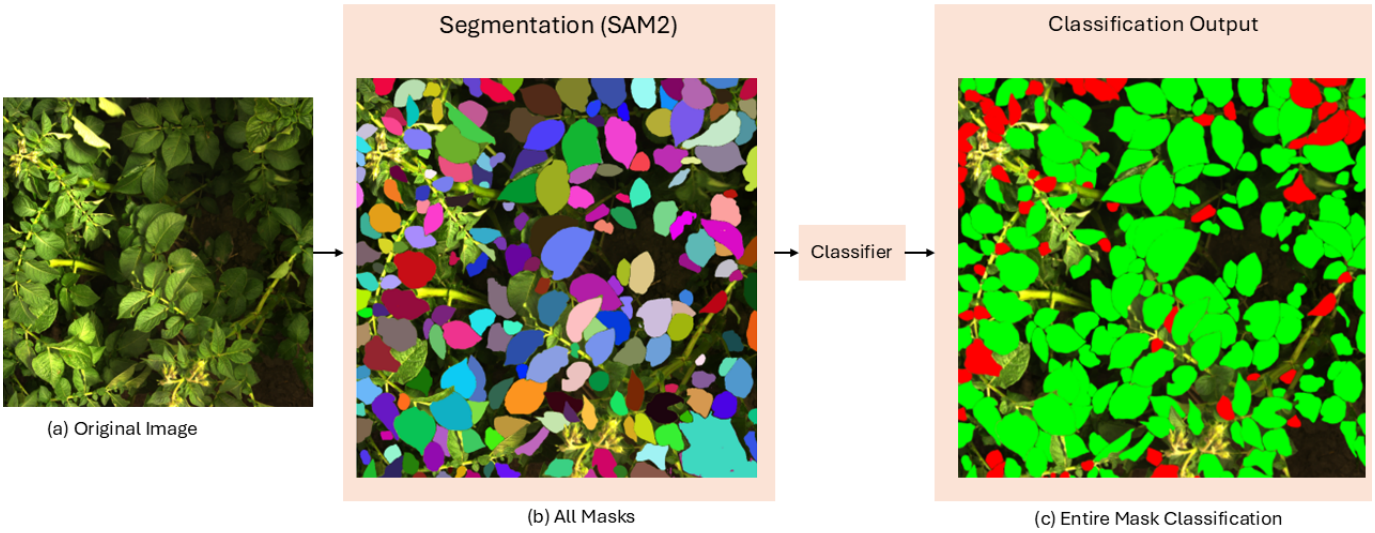


Fig. 6: **Pipeline Overview**: The workflow of the proposed two-stage mechanism, illustrating the steps from data input to the final prediction. (a) Original image used for segmentation and classification. (b) Masks generated by SAM2's automatic mask generator. (c) Entire image classification overlay using a pre-trained classifier.

known for its computational efficiency and lightweight architecture, employs convolutional layers with residual blocks to extract meaningful mid-level features, making it practical for resource-constrained environments. Its final fully connected layer was adapted for binary classification, focusing on texture patterns linked to leaf health. EfficientNet-B0, with its scalable and lightweight design incorporating depthwise separable convolutions and attention-based features, was particularly effective for high-resolution images while maintaining computational efficiency.

All models' final classification layers, originally designed for multi-class tasks, were modified to output probabilities for the two target classes, ensuring their suitability for this binary classification task.

3.5 Pipeline: Combining SAM2 and Classifier

This study integrates SAM2 and a classifier into a two-stage pipeline for classifying potato leaf health and assessing overall plant health. The workflow, as illustrated in 6, begins with the input of the original image, which serves as the basis for segmentation and classification.

SAM2 then using its automatic mask generation feature 3.3, identifies and segments multiple regions in the image without requiring explicit prompts. This functionality accelerates the segmentation process and aids in detecting unannotated regions, providing bounding boxes that can be reviewed and labeled by domain experts.

Once segmentation is complete, the pipeline transitions to the classification stage, where all segmented regions generated by SAM2 are classified using a pre-trained classifier. Each region is individually analyzed and categorized as either "Healthy" or "PVY-infected," ensuring precise evaluation of every segmented mask. This classification stage focuses on leveraging the pre-trained classifier's capabilities to provide detailed health predictions for all segmented regions.

To determine the overall health of the plant, the pipeline employs a majority voting mechanism. If the majority of segmented regions are classified as "PVY," the plant is labeled as "PVY-infected", otherwise, it is labeled as "Healthy." This dual-level assessment, both at the segment and plant levels, ensures a comprehensive evaluation. The pipeline's outputs include not only classification labels for specific regions of interest but also visualizations of segmented regions with their respective classifications, making it a streamlined approach for plant health assessment.

3.6 Feature Extraction

In this study, feature extraction was employed to utilize reference images as guides for classification. By leveraging classifier models such as Vision Transformer (ViT-B/16), ResNet-18, and EfficientNet-B0, we extracted feature representations that serve as compact and semantically rich embeddings of the reference images. These models were fine-tuned on our custom PVY dataset, which includes images of healthy and PVY-infected potato leaves. This fine-tuning ensured that the models adapted their learned features to the specific characteristics of the PVY dataset, enhancing their relevance and effectiveness for our application.

The feature extraction process involved passing reference images through these fine-tuned models and capturing the activations from the final feature layer. This layer, being the last before classification, encapsulates the network's hierarchical understanding of the image, combining low-level visual features with high-level semantic concepts. These extracted features provided a high-dimensional representation of the image content in a compressed format.

By extracting features from the reference images, we enabled a static similarity measure in the feature space. This allowed us to evaluate the alignment between the reference image and input data by comparing their feature embeddings. The similarity was computed using standard metrics such as Euclidean distance, ensuring a consistent and interpretable measure of closeness in the semantic space as discussd in the Section 3.7.

This approach was integral to Experiment 4.2, which investigates the impact of reference image guided techniques. The extracted reference features provided the foundation for assessing how well the input aligns with the semantic information encoded in the reference images.

By integrating this feature extraction process into our methodology, we established a reliable framework for utilizing reference images as a classification guide, enabling effective comparison and analysis in subsequent experimental stages.

3.7 Similarity Computation

Similarity computation evaluates the alignment of feature embeddings within the embedding space, crucial for our classification task. This subsection introduces Euclidean distance as the primary metric.

3.7.1 Euclidean Distance

Euclidean distance measures the straight-line distance between two feature vectors in the embedding space. Mathematically, it is defined

as:

$$d_E(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{4}$$

where x and y are the feature vectors of two images, $x = [x_1, x_2, \ldots, x_n]$ and $y = [y_1, y_2, \ldots, y_n]$, and n is the dimensionality of the vectors. Euclidean distance quantifies the overall dissimilarity between two vectors, with smaller values indicating greater similarity. This metric emphasizes magnitude differences in the embedding space.

By utilizing Euclidean distance, we quantify the global differences in magnitude between feature vectors, which is particularly effective for identifying significant variations in visual patterns between healthy and PVY-infected leaf regions. This approach ensures accurate similarity computations, enabling reliable differentiation of infected areas during downstream tasks.

3.8 Hyperparameters

For this study, Vision Transformer (ViT-B/16), ResNet-18, and EfficientNet-B0 were employed. The training procedure utilized gradient descent, guided by the cross-entropy loss function to evaluate model performance against ground truth labels. For this binary classification problem, the cross-entropy loss, defined in Equation 5, was employed. Here, $p(x)$ denotes the ground truth label, and $q(x)$ represents the network's predicted probability:

$$\text{Cross Entropy Loss}(p,q) = -\sum_{x \in \text{classes}} p(x) \cdot \log q(x). \tag{5}$$

The training process was configured with the following hyperparameters:

- **Number of Epochs:** 20
- **Learning Rate:** 3×10^{-5}
- **Weight Decay:** 1×10^{-4}
- **Batch Size:** 16

To ensure efficient training and prevent overfitting, early stopping was incorporated as a regularization technique. Specifically, an epoch was defined as a complete sweep over the entire training dataset, where all samples were processed once. Training was early stopped when the validation loss failed to improve for five consecutive epochs. This approach, combined with the hyperparameter choices, optimized training time while maintaining robust model generalization across all evaluated architectures.

3.9 Evaluation Metrics

To comprehensively evaluate our pipeline, we employ distinct metrics tailored to different tasks. Intersection over Union (IoU) is used to assess segmentation quality, specifically the alignment between predicted bounding boxes and ground truth annotations in segmentation tasks. Meanwhile, classification tasks are evaluated using metrics derived from True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN), including precision, recall, accuracy, and the F1-score.

3.9.1 Metrics for Segmentation Evaluation

IoU is a key metric for evaluating the overlap between the predicted bounding boxes and the ground truth annotations. It is defined in the Equation 6:

$$\begin{aligned} \text{IoU} &= \frac{\text{Area of Overlap}}{\text{Area of Union}} \\ &= \frac{\text{Area of (Predicted Box} \cap \text{Ground Truth Box)}}{\text{Area of (Predicted Box} \cup \text{Ground Truth Box)}}. \end{aligned} \tag{6}$$

IoU values range from 0 to 1, where 1 indicates perfect overlap and 0 signifies no overlap. An IoU threshold, often set at 0.7, is commonly used to determine True Positives (*TP*) in object detection tasks.

In our experimentation, IoU is critical for evaluating the performance of SAM2 in segmentation tasks. By quantifying the alignment between predicted and ground truth bounding boxes, we assess the efficacy of various prompting strategies—point, box, and edge—in guiding the segmentation process. This evaluation plays a vital role in analyzing the impact of prompts on the accuracy in the Experiment 4.3.

3.9.2 Metrics for Classification Evaluation

Classification tasks are evaluated using metrics derived from TP, TN, FP, and FN. These metrics provide a comprehensive view of model performance, particularly in distinguishing between healthy and PVY leaves. Below are the key metrics and their formulations:

Accuracy: Measures the overall proportion of correct predictions, defined as:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \tag{7}$$

Precision: Indicates the ratio of correctly predicted PVY leaves to all leaves predicted as PVY, emphasizing the minimization of false positives:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{8}$$

Recall: Also known as sensitivity, represents the proportion of PVY leaves correctly identified, focusing on minimizing false negatives:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{9}$$

F1-score: Provides a harmonic mean of precision and recall, offering a balanced view of both metrics:

$$F_1\text{-score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{10}$$

These metrics are particularly crucial for assessing classification models where the misclassification of diseased leaves as healthy is more detrimental than the reverse.

In this study, accuracy, precision, recall, and the F1-score collectively offer a detailed understanding of the classifiers' ability to detect and predict healthy and PVY leaves effectively. By using these metrics, we ensure that the performance of our classification models is robust and interpretable across different experimental setups.

4 Experiments & Results

The following section presents the experiments conducted, each addressing a specific aspect of the pipeline, along with their associated results. In Experiment 1, we investigate plant per-leaf classification, focusing on the impact of removing blurry images from the dataset and analyzing how this preprocessing step affects classification performance. In Experiment 2, we examine different techniques for classifying test samples using feature embeddings derived from the pre-trained models discussed in Section 3.4. Finally, Experiment 3 delves into the use of varied prompt types, exploring their significance in dense image segmentation and their influence on model generalization.

4.1 Experiment 1: Impact of Blurry Image Removal on Plant Leaf Classification

In this experiment, we introduced a preprocessing step in the data loading phase of the training pipeline. While loading images from disk, we calculated the variance for each box annotation or the leaf itself using Laplacian method, as described in Section 3.2.2. To systematically filter blurry images, we fit a normal distribution to the calculated variance values. By leveraging the standard deviation of the distribution, we dynamically adjusted the blur threshold, allowing us to statistically exclude a quantitative subset of blurry images.

The amount of data considered for each standard deviation offset value, along with the number of blurred samples excluded, is shown

in the Table 2. These statistics are averaged across k=3 folds during cross-validation.

Here, a standard deviation of -2.0 is used as the baseline, representing all data samples without excluding any blurred images.

Table 2: Dataset statistics across average of k = 3 folds with varying blur levels using the mean and standard deviation of the distribution. The table shows the number of healthy and PVY samples for train, validation, and test splits, along with total samples and blurred samples removed.

Blurred Dataset Statistics Across Average of k = 3 Folds								
Std.	**Train**		**Val**		**Test**		**Total**	**Blurred**
Dev.	**Healthy**	**PVY**	**Healthy**	**PVY**	**Healthy**	**PVY**	**Samples**	**Removed**
-2.0 (Baseline)	652	577	326	288	198	250	2291	0
-1.0	580	537	290	269	198	250	2124	167
-0.75	494	467	247	233	198	250	1889	402
-0.50	397	391	199	196	198	250	1631	660
-0.25	320	331	160	166	198	250	1425	866

The performance metrics for Vision Transformer (ViT-B/16), ResNet-18, and EfficientNet-B0 under varying levels of blur (denoted by the standard deviation of the blur applied) are presented in Table 3. The table shows that as the level of increase in removal of the blur samples (from standard deviation -2.0 to -0.25), the overall performance of all three models generally declines, with noticeable drops in Accuracy, Precision, Recall, and F1 Score. Therefore, upon closer analysis, it becomes evident that the primary factor influencing model performance in this diverse agricultural setup is the availability of data, not the degree of blur.

Among the three models, ViT-B/16 consistently demonstrates the highest performance across all levels of blur. For instance, at Std. Dev. -2.0, ViT achieves an Accuracy of 74.70%, a Precision of 83.50%, and an F1 Score of 75.12, significantly outperforming both ResNet-18 and EfficientNet-B0. Even at removing most of the blur samples (Std. Dev. -0.25), ViT-B/16 maintains relatively stable performance, achieving an F1 Score of 70.12 compared to 65.35 and 62.34 for ResNet-18 and EfficientNet-B0, respectively. These results highlight ViT-B/16's robustness in handling degraded image quality. However, it is crucial to note that ViT-B/16's performance is most influenced by the diversity and size of the dataset rather than the level of blur.

Interestingly, the observed trends in Precision and Recall suggest that blur itself is not the primary driver of performance variation. Precision remains consistently higher than Recall across all models and blur levels, indicating that the models prioritize reducing false positives over capturing all true positives. For example, at Std. Dev. -2.0, ViT-B/16 achieves a Precision of 83.50% but a Recall of 68.40%, a pattern mirrored in ResNet-18 and EfficientNet-B0. This Precision-Recall trade-off further underscores that the data itself, including its diversity and volume, is the most significant factor impacting performance. Additionally, this implies that the model's higher Precision reduces the risk of incorrectly classifying healthy leaves as PVY. This, in turn, helps avoid unnecessary treatment of healthy leaves, such as the unwarranted application of pesticides or even the premature removal of the plant itself.

Upon closer observation, we notice an increase in Recall for both ResNet-18 and EfficientNet-B0 when blurry samples are removed at standard deviations of -0.75 and -0.5, respectively. Specifically, ResNet-18 shows an improvement in Recall from 56.93% to 63.33%, while EfficientNet-B0 increases from 62.27% to 65.33%. This indicates that clearer images with sharper details lead to a reduction in False Negatives, where PVY-infected leaves are mistakenly predicted as healthy. This reduction is critical for our use case, as it helps prevent the spread of PVY and mitigates the significant losses associated with undetected infections.

In the context of agricultural setups, where images often contain diverse and have complex backgrounds, it becomes clear that expanding the dataset with varied and representative samples is paramount. The highest accuracies are observed when the full dataset is employed, confirming that model performance in this problem domain is more data-dependent than blur-dependent. This finding emphasizes the need to prioritize dataset augmentation and diversification over pre-processing techniques like filtering out blurry samples for addressing performance limitations.

In conclusion, the results in Table 3 demonstrate that while ViT-B/16 outperforms ResNet-18 and EfficientNet-B0 in handling blur, the critical factor for achieving high accuracy in this agricultural setup is the availability of comprehensive and diverse data. Blur has a negligible impact compared to the dataset itself, making data expansion a central strategy for improving model performance.

4.2 Experiment 2: Impact of Reference Image based Guidance

In this experiment, we compare three techniques for classifying test samples using feature embeddings derived from the pre-trained models described in Section 3.4. The methods—Single Feature Embedding Distance Comparison, Averaged Feature Embedding Distance Comparison, and Majority Voting—leverage spatial relationships in a high-dimensional embedding space to perform classification. To ensure consistency, Euclidean distance is used as the metric across all methods, as detailed in Section 3.7.1.

We utilize the dataset outlined in Section 3.2 to compare results with Experiment 4.1. Unlike the previous experiment, predictions are not performed directly by the classifier. Instead, they are based on embeddings extracted from the samples, offering a different perspective on the classification process. This experiment uses the entire dataset, which is the baseline, without excluding blurred samples. The dataset is divided into reference, validation, and test subsets to facilitate the evaluation.

4.2.1 Single Feature Embedding Distance Comparison

The first approach, Single Feature Embedding Distance Comparison, utilizes the proximity of feature embeddings in the embedding space for classification. For a given test image and a reference image, per-leaf feature embeddings are extracted, and the distance between the two embeddings is calculated. Classification is then determined by comparing this distance against a dynamically computed threshold, which serves as the decision boundary, as illustrated in the Figure 7.

The threshold is computed by leveraging validation data through an iterative process. Initially, distances between validation embeddings and reference embeddings are calculated. For each reference embedding, the minimum and maximum distances are precomputed to enable normalization. Normalizing the distances accounts for variations in embedding scales, ensuring consistent comparisons across the dataset.

Next, a range of potential thresholds, typically spanning 0.0 to 1.0, is evaluated. Predictions are generated for validation embeddings at each threshold based on their normalized distances to reference embeddings. Metrics such as accuracy, precision, recall, and F1-score are calculated for each threshold and aggregated across all reference embeddings.

The optimal threshold is determined as the one that maximizes the F1-score, a metric chosen for its ability to balance precision and recall. This data-driven optimization tailors the threshold to the specific characteristics of the dataset, accounting for class overlaps and distributional variability.

During testing, the precomputed minimum and maximum distances from the validation phase are reused to normalize test embeddings, ensuring that the threshold remains effective on unseen data. This normalization allows the approach to adapt to variations in embedding distributions, maintaining consistency.

Using this dynamically calculated threshold, a test sample is classified as belonging to the same label as the reference embedding

Table 3: **Performance Metrics**: Comparison of ViT-B/16, ResNet-18, and EfficientNet-B0 models under different levels of blur, where -2.0 represents no blur. The table includes Accuracy, Precision, Recall, and F1 Score for each model (in %).

Std. Dev. of Blur	**ViT-B/16**				**ResNet-18**				**EfficientNet-B0**			
	Accuracy (in %)	Precision (in %)	Recall (in %)	F1 Score (in %)	Accuracy (in %)	Precision (in %)	Recall (in %)	F1 Score (in %)	Accuracy (in %)	Precision (in %)	Recall (in %)	F1 Score (in %)
-2.0 (Baseline)	**74.70**	**83.50**	**68.40**	**75.12**	68.53	81.05	56.93	66.77	70.09	79.81	62.27	69.83
-1.0	72.92	82.34	65.60	72.94	69.94	82.15	58.93	68.60	68.38	81.07	56.80	66.64
-0.75	70.39	81.02	61.20	69.60	70.98	80.46	63.33	70.83	71.21	80.98	63.47	70.97
-0.50	68.60	76.84	62.53	68.86	64.14	72.41	58.67	64.51	68.45	75.38	65.33	69.32
-0.25	69.57	77.20	64.53	70.12	64.73	73.15	59.73	65.35	63.24	72.15	56.00	62.34

Table 4: **Performance Metrics**: Comparison of ViT-B/16, ResNet-18, and EfficientNet-B0 models under different embedding comparison techniques. The table includes Accuracy, Precision, Recall, and F1 Score for each model (in %).

Comparison Techniques	**ViT-B/16**				**ResNet-18**				**EfficientNet-B0**			
	Accuracy (in %)	Precision (in %)	Recall (in %)	F1 Score (in %)	Accuracy (in %)	Precision (in %)	Recall (in %)	F1 Score (in %)	Accuracy (in %)	Precision (in %)	Recall (in %)	F1 Score (in %)
Single Feature Embedding	61.16	68.81	55.60	61.50	63.39	70.48	59.20	64.35	62.72	71.07	56.00	62.64
Averaged Feature Embedding	70.76	73.15	75.20	74.16	61.83	62.15	80.80	70.26	70.31	69.06	84.80	76.12
Majority Voting	74.11	75.77	78.80	77.25	70.09	66.11	**95.20**	78.03	70.09	66.11	**95.20**	78.03

if the distance is below the threshold. Otherwise, it is assigned the opposite label. This method provides an efficient and robust classification mechanism, leveraging spatial embedding relationships for decision-making.

4.2.2 Averaged Feature Embedding Distance Comparison

Averaged Feature Embedding Distance Comparison builds upon the Single Feature Embedding Distance Comparison method 4.2.1 by introducing an additional step to compute a mean reference embedding. This approach aims to reduce the influence of individual variations in embeddings and provide a more generalized representation of the class.

In this method, we specify both the class label and the number of reference images. Since, we are interested to detect the PVY samples, we give the class label corresponding to PVY class and number of reference image as five. From the reference dataset, five samples of PVY class are selected in a random fashion, and their feature embeddings are extracted. These embeddings are then averaged to compute a single mean reference embedding for the class. This mean embedding represents a consolidated feature space for the PVY class, effectively smoothing out outliers or unique variations present in individual embeddings.

For classification, the distance between the test embedding and the mean reference embedding is calculated. We use the threshold already optimized in the Single Feature Embedding Distance Comparison Experiment 4.2.1. This reuse of thresholds ensures a fair comparison between the two approaches, as it isolates the effect of averaging embeddings on classification performance.

Much like the previous method 4.2.1, if the distance is below the threshold, as our reference class is PVY, we predict the sample PVY else Healthy.

4.2.3 Majority Voting

The Majority Voting method does classification by considering the top-k nearest reference embeddings for each test sample. Instead of relying on a single reference embedding, this method aggregates decisions from multiple neighbors to provide a more consensus-driven classification.

The process begins by computing the distance between the test embedding and all reference embeddings. These distances were then sorted to identify the top-k nearest neighbors in the embedding space. The corresponding labels of these top-k reference embeddings are extracted, and a majority vote was conducted to determine the predicted label for the test sample. This approach leverages the collective agreement among the nearest neighbors to make a decision, reducing the impact of outliers or noise in individual embeddings.

To optimize the parameter k, a validation-based dynamic selection process was employed. A range of k values, typically starting from 50 and in increasing steps of 50 until max length of validation set samples, is evaluated on the validation dataset. For each k, predictions are generated for the validation embeddings, and classification metrics such as accuracy, precision, recall, and F1-score are computed. The optimal k is selected as the value that maximizes the F1-score, balancing precision and recall. This dynamic optimization ensures that the majority voting process adapts to the dataset's specific characteristics, such as the density of embeddings or class distributions.

During testing, the optimized k was used to classify unseen test samples. The distances between each test embedding and the reference embeddings are calculated, and the majority vote was conducted among the top-k neighbors to predict the label. By incorporating multiple neighbors in the decision-making process, this method provides a robust classification mechanism that accounts for the variability and diversity within the reference embeddings.

The Majority Voting method aims to evaluate whether aggregating decisions from multiple embeddings improves classification performance compared to single embedding-based methods, such as the previous methods 4.2.1 and 4.2.2. This experiment explores the strength of ensemble-like techniques in embedding-based classification, offering a more generalized approach for decision-making in high-dimensional feature spaces.

From the Table 4, we understand that under the Single Feature Embedding technique, ResNet-18 achieved the highest Precision (70.48%), indicating its strong ability to minimize false positives. However, its relatively lower Recall (59.20%) highlights its limitation in detecting all positive instances, suggesting underestimation of PVY leaves. EfficientNet-B0 displayed the best Precision among the models (71.07%), demonstrating robust positive prediction capability. In contrast, ViT-B/16 exhibited both lower Precision (68.81%) and Recall (55.60%), which indicates struggles in both accurate and complete detection of PVY leaves. Overall, the Single Feature Embedding technique showed limited effectiveness due to its reliance on single-feature representations, which may lead to incomplete detections and CNN based architectures like ResNet and Efficient models outperfoming compared to vision transformer models like ViT.

The Averaged Feature Embedding technique significantly improved Recall across the models by leveraging multiple reference embeddings. EfficientNet-B0 achieved the highest Recall (84.80%),

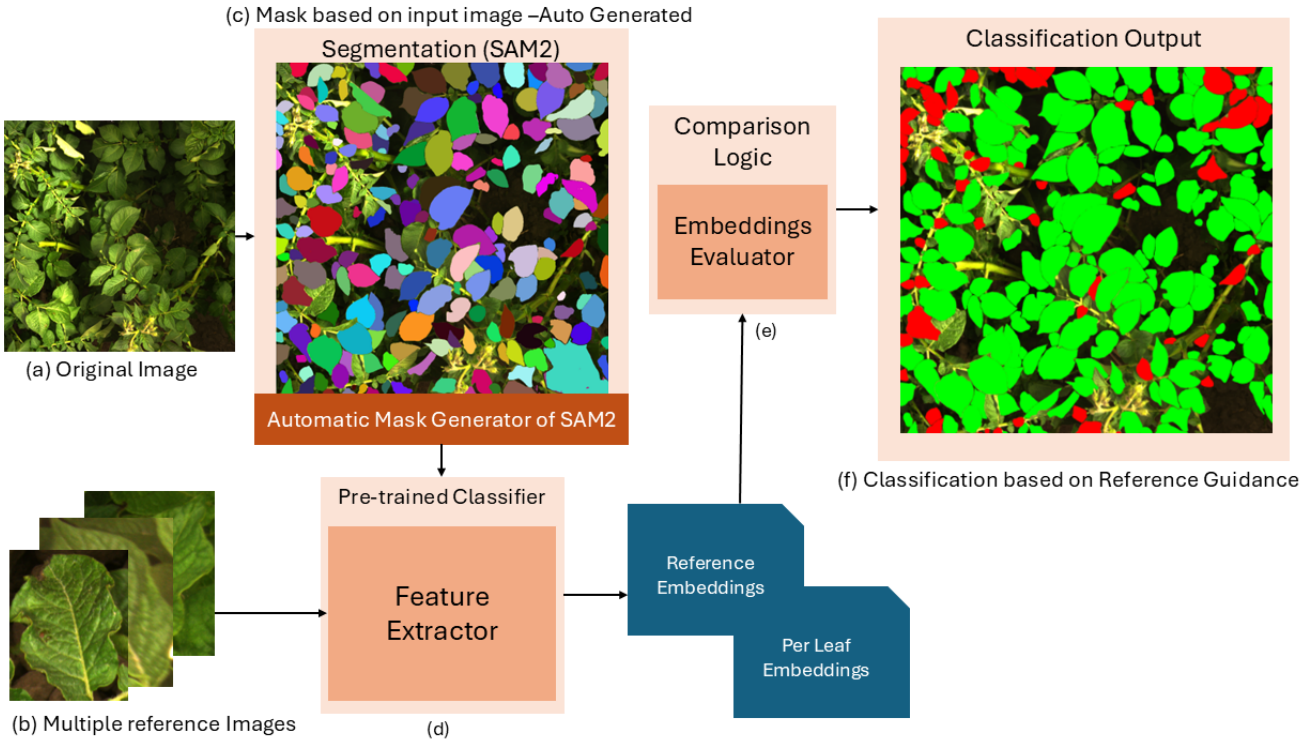


Fig. 7: **Reference Image-Guided Workflow**: A workflow illustrating the process from data input to the final prediction, using a reference image as a guide for classification. (a) Original image used for segmentation and classification. (b) Multiple reference images provided as input. (c) Masks generated by the Automatic Mask Generator of SAM2. (d) Feature extraction logic using a pre-trained classifier. (e) Distance comparison of feature embeddings. (f) Classification results based on reference images, with green masks indicating healthy leaves and red masks representing PVY-infected leaves.

demonstrating superior sensitivity in detecting PVY leaves, though it experienced a slight trade-off in Precision (69.06%). Similarly, ResNet-18 showed a substantial improvement in Recall (80.80%), though Precision remained moderate (62.15%), making it suitable for applications like ours, whose intent is to avoid false negatives by emphasizing sensitivity over specificity. ViT-B/16, while achieving the best Precision (73.15%) among the models, demonstrated slightly lower Recall (75.20%) than EfficientNet-B0.

It is worth noting that smoothing representations primarily impacts Recall, with a consistent increase of around 20% across all three models. Precision, however, fluctuates between models: ViT sees an increase of 4.34%, while ResNet and EfficientNet experience reductions of 8.33% and 2.01%, respectively. These results suggest that averaging embeddings enables models to perform better in scenarios where sensitivity is critical.

The Majority Voting technique yielded the highest Recall across all models, particularly for ResNet-18 and EfficientNet-B0, both achieving a 95.20%, significantly reducing false negatives in PVY detection. However, this high sensitivity came at the cost of reduced Precision (66.11%), indicating a tendency to overpredict positive instances. ViT-B/16, while maintaining a balanced Recall (78.80%), demonstrated better Precision (75.77%) than the other two models, showcasing its ability to balance sensitivity and specificity effectively. The consistent improvement in Recall across models using Majority Voting highlights its suitability for detecting PVY leaves in scenarios where missing detections could have severe consequences. Notably, the similarity in results between ResNet-18 and EfficientNet-B0 can be attributed to the selection of an optimal k-value, which differed from that chosen by ViT-B/16 due to their CNN-based architecture.

Overall, the results indicate that ViT-B/16 outperforms the other models in terms of balancing Precision and Recall, especially under the Majority Voting technique. EfficientNet-B0 demonstrated superior Recall under both Averaged Feature Embedding and Majority Voting, making it a vital for applications prioritizing sensitivity in PVY detection. ResNet-18, despite lower Precision, showed remarkable improvement in Recall using Averaged Feature Embedding and Majority Voting, making it suitable for scenarios where minimizing false negatives is critical.

In conclusion, the choice of evaluation technique plays a pivotal role in determining the trade-offs between Precision and Recall in PVY detection. Majority Voting consistently enhances Recall, making it ideal for applications requiring high sensitivity. Averaged Feature Embedding provides a balanced alternative, achieving high Recall while maintaining competitive Precision. These insights highlight the importance of selecting techniques tailored to the specific requirements of PVY detection, particularly when managing the trade-offs between false positives and false negatives.

4.3 Experiment 3: Impact of Prompting Techniques on Instance Segmentation

This experiment evaluates the effectiveness of three distinct prompting techniques—point-based, box-based, and edge-based—for instance segmentation tasks. The primary objective is to assess how each technique influences segmentation accuracy, focusing on the Intersection over Union (IoU) 3.9 as the primary metric. By studying these techniques, the experiment aims to develop a deeper understanding of prompt heuristics and their impact on segmentation performance.

We utilize the dataset outlined in Section 3.2 Prompts are generated based on the specific characteristics of each technique, fed into the SAM-based segmentation model, and evaluated against ground truth bounding boxes. This unified approach ensures a consistent and fair comparison across all techniques.

For prompting, we provide the original plant image along with

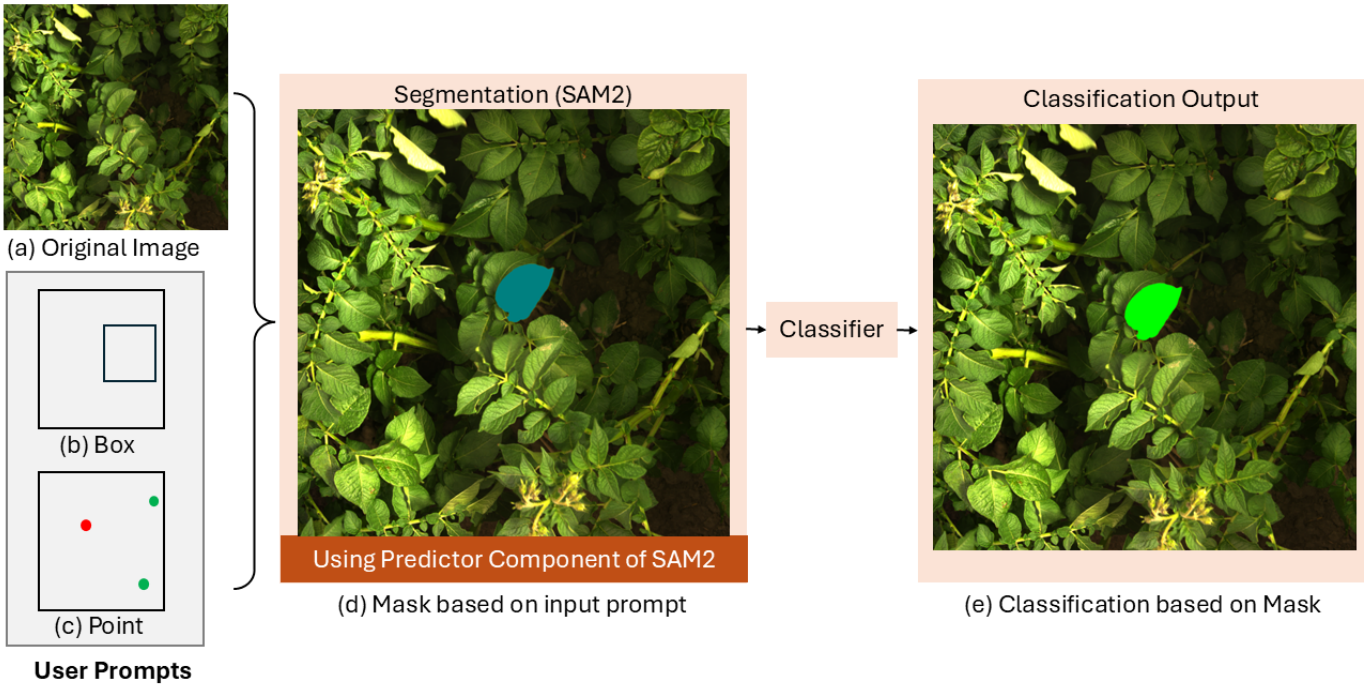


Fig. 8: **Prompting Workflow**: A workflow illustrating the process from data input to the final prediction, guided by user prompts for segmentation and classification. (a) Original image used for segmentation and classification. (b) Point prompt. (c) Box prompt. (d) Mask(s) generated by SAM2's predictor component. (e) Image classification overlay produced by a pre-trained classifier based on the user prompt.



(a) Point Prompt (b) Edge Point Prompt (c) Box Prompt

Fig. 9: **Different Prompt Techniques**: Black indicates the user-defined prompts (point or box). The green box represents the ground truth, while the red box denotes the segmentation prediction.

user-defined prompts, such as points or boxes, to SAM2's predictor component. This component generates the corresponding segmentation mask and bounding box for the leaf based on the user prompt. The segmented leaf sample is then passed to the pre-trained classifier, which outputs the classification result with an overlay, as illustrated in the Figure 8.

Point-based prompting involves generating prompts using the center points of bounding boxes derived from ground truth YOLO annotations. These central points act as prompts for the SAM-based model, which predicts segmentation masks and bounding boxes based on them. This technique is computationally efficient and easy to implement, making it suitable for scenarios where objects are well-centered and have regular shapes.

In box-based prompting, bounding box prompts are generated by expanding the ground truth YOLO bounding boxes using a random threshold, specified in pixels by the user at runtime, within a range of ten to one hundred. This method aims to incorporate additional spatial context surrounding the objects.

Table 5: Comparison of average IoU scores for Point, Box, and Edge prompting techniques.

Average IoU Scores for Different Prompt Techniques

Prompt Technique	**Average IoU (%)**
Point	86.00
Box	76.00
Edge	31.87

Edge-based prompting focuses on extracting edge points from leaf images within the ground truth YOLO bounding boxes using the Sobel gradient magnitude, as shown in Figure 9 (b).. Among the

extracted points, the one with the highest gradient intensity is selected as the prompt. This technique is expected to excels at capturing object boundaries, making it particularly effective for irregularly shaped objects.

For each technique, up to one hundred prompts were generated per image, and the corresponding segmentation results were evaluated quantitatively and qualitatively. Quantitative metrics included the average IoU across all hundred generated prompts as shown in the Table 5. Qualitative analysis involved visualizing predicted bounding boxes, overlaid ground truth annotations, and the user input prompt as shown in the Figure 9. This comprehensive evaluation provided an intuitive understanding of the strengths and weaknesses of each technique under different scenarios.

From the Table 5, it is evident that point-based prompting provides the most accurate predictions with an average IoU of 86%. However, some bounding boxes span the entire leaf, leading to inconsistencies with ground truth annotations, which vary in size. Box-based prompting, with an average IoU of 76%, extend the bounding box by including additional context and encounters IoU mismatches in dense images due to overlapping regions with other leaves. Edge-based prompting, which achieves an average IoU of 31.87%, struggles in dense images as it often confuses adjacent leaves with the target object.

These findings are also visualized in the Figure 9, highlighting how each technique performs under various scenarios. This analysis underscores the need for selecting or adapting prompting techniques based on the specific dataset and segmentation task at hand.

5 Discussion, Conclusion & Future Work

This chapter sums up the paper by interpreting the results, formulating conclusions and proposing future work.

5.1 Discussion

In summary, this study encompasses three experiments aimed at enhancing plant leaf classification and segmentation: the impact of blurry image removal, the use of reference images as guide for classification, and the evaluation of prompting techniques/heuristics for instance segmentation.

In the first experiment, the performance of ViT-B/16, ResNet-18, and EfficientNet-B0 models was evaluated under varying levels of blur, represented by the standard deviation of blur values. The findings demonstrate distinct trends across models, emphasizing the influence of image clarity on precision and recall metrics in detecting Potato Virus Y (PVY).

The results underscore that while ViT-B/16 is less impacted by image clarity and performs reliably in complex agricultural datasets, ResNet-18 and EfficientNet-B0 are more sensitive to blur, benefiting from a careful balance between data volume and quality. These findings suggest that for PVY detection in diverse agricultural contexts, prioritizing models with strong generalization capabilities, such as ViT-B/16, is crucial. Furthermore, dataset augmentation and diversity appear to play a more significant role than preprocessing techniques like blur removal in maintaining high detection accuracy and recall. This highlights the need for robust training pipelines that account for the inherent variability in agricultural imagery.

The second experiment evaluated the use of Majority Voting, Averaged Feature Embedding, and Single Feature Embedding techniques to classify leaves as healthy or PVY, focusing on the trade-offs between precision and recall. Majority Voting consistently demonstrated the highest recall, particularly for ResNet-18 and EfficientNet-B0, making it highly effective in minimizing false negatives. However, its tendency to overpredict positive instances highlighted a trade-off in precision. Averaged Feature Embedding showcased strong adaptability, striking a balance between precision and recall, particularly for ViT-B/16, which excelled in generalization. Single Feature Embedding, while effective in scenarios with closely aligned reference and target images, struggled to achieve comparable generalization. The robust recall performance of Majority Voting and the balanced adaptability of Averaged Feature Embedding underscore their potential for inclusion in modular classification pipelines for agricultural applications where both sensitivity and specificity are critical.

When comparing the two Experiments 4.1 and 4.2, the results suggest that while ViT-B/16 remains the most stable model under both blurred conditions and embedding techniques. Its performance is notably enhanced with Majority Voting, where it achieves the highest precision (75.77%) and a balanced recall (78.80%), showcasing its ability to maintain sensitivity and specificity. ResNet-18 and EfficientNet-B0, despite their higher sensitivity to image blur, demonstrate significant improvements with Majority Voting, where their recall is substantially enhanced, effectively mitigating their susceptibility to false negatives. For scenarios emphasizing high recall, such as early disease detection, Majority Voting offers a reliable alternative, particularly for CNN-based models. These findings underscore the importance of tailoring model selection and embedding techniques to the specific challenges posed by agricultural datasets. Robust training pipelines incorporating diverse augmentation strategies and embedding techniques can further enhance detection accuracy and recall in real-world conditions.

The third experiment investigated prompting techniques for instance segmentation, comparing point-based, box-based, and edge-based approaches. Point-based prompting achieved high IoU values, offering accurate predictions but occasionally over-segmenting leaves, resulting in predictions that span the entire leaf, despite annotation size variations. Box-based prompting demonstrated improved segmentation accuracy by leveraging spatial context. However, in dense images, overlapping regions within the user-prompted box led to IoU mismatches between predictions and ground truth. Edge-based prompting, while adept at handling irregularly shaped objects, struggled with dense images, often confusing adjacent leaves, and proved sensitive to noise in low-quality inputs. These findings underscore the importance of tailoring prompting techniques to the dataset and segmentation task, balancing precision, contextual awareness, and the ability to handle image complexity.

5.2 Conclusion

This study presents a comprehensive approach to enhancing plant leaf classification and instance segmentation in agricultural datasets characterized by diverse and complex imagery. The findings emphasize that dataset diversity and size significantly influence classification performance, outweighing the impact of preprocessing techniques like removing out-of-focus or blurry samples from the dataset. Among the tested models, ViT-B/16 demonstrated exceptional robustness to image quality variations, consistently outperforming ResNet18 and EfficientNet-B0.

Embedding-based classification strategies, particularly majority voting, proved to be highly effective. This approach highlights the power of ensembling methods, which combine predictions from multiple reference embeddings to enhance robustness and accuracy. It also enhances user-friendliness by providing end-users with control over the classification process, as they can guide the system with reference images rather than relying solely on a black-box classification of full-resolution inputs. Additionally, it facilitates generalization to potato cultivars by eliminating the need for fine-tuning the model for each specific cultivar, making the system more versatile and accessible. This modularity makes the approach highly adaptable and reduces maintenance requirements.

From a real-world perspective, these findings are particularly relevant to stakeholders in precision agriculture, including farmers, agronomists, and researchers. The ability to classify and segment plant leaves with high accuracy enables early detection of plant diseases like PVY, which can have significant economic and environmental impacts. For farmers, this system can streamline the process of monitoring crop health, reducing labor-intensive manual inspection and enabling targeted interventions. For agronomists and researchers, the modular design of the system allows for easy adaptation to other crops, fostering broader applications across the

agricultural sector.

To address the research question of whether prompt-based models can effectively segment individual leaves and, in combination with a fine-tuned classifier, classify healthy and PVY-affected leaves, the study demonstrates that integrating reference image comparison techniques into a prompt-based framework provides a viable solution. This approach ensures that the models remain robust to variations in crop cultivars, environmental conditions, and image quality, making them applicable across diverse agricultural scenarios.

For segmentation tasks, the study revealed that in dense and complex agricultural images, point-based prompting techniques generally outperform box-based and edge-based prompts. Additionally, by utilizing models like SAM2, which generate automatic segmentation masks and bounding boxes without requiring fine-tuning, the process of annotation becomes significantly more efficient. Passing these automatically generated masks and bounding boxes to a classifier enables subject matter experts to quickly and easily annotate and review the results. This streamlined process not only accelerates data generation but also ensures greater accuracy, facilitating the creation of high-quality labeled datasets for complex agricultural scenarios.

Qualitative analysis further underscores the practical utility of this approach. For instance, the ability to isolate and classify specific leaves in complex scenes empowers users to assess plant health at a granular level, enabling precision interventions such as localized pesticide application or nutrient management. Moreover, the improved efficiency in annotation and review processes not only saves time but also accelerates the creation of additional data for fine-tuning models. This, in turn, supports more informed decision-making, reduces costs, and promotes sustainable farming practices.

In summary, this research highlights the effectiveness of embedding-based classification guided by reference images in addressing challenges in agricultural image analysis. The decoupled approach, combining segmentation and classification, demonstrates the potential to improve accuracy and usability in complex scenarios. Moreover, the decoupled approach makes it easier to replace the classifier layers and independently optimize them for better predictions.

5.3 Future Work

This section outlines specific areas where the current methodology can be extended to address key challenges in agricultural analysis.

First, while this study has primarily focused on extracting features from the last layer of the classifier, future work could explore leveraging feature embeddings from intermediate layers. These layers may capture additional, distinguishable features, offering a richer representation of the data and potentially improving classification outcomes. Additionally, experimenting with different classifier backbones and integrating more complex classification models could further enhance the system's performance by capturing nuanced patterns and relationships within the data.

Second, improving model generalization is a critical area for future work. While this study focused on potato plants, future research should test the methodology on different species of potato plants, various potato cultivars, or even crops beyond potatoes. Developing a generalized model capable of detecting diseases across multiple crops would make the approach more versatile and impactful in diverse agricultural scenarios. Addressing class imbalances in the dataset as seen in the Table 1 is another key aspect to enhance model performance. Techniques such as weighted sampling could be employed to ensure a more balanced representation of classes during training. Additionally, incorporating synthetic data generation could significantly enhance model training, particularly in scenarios with limited labeled datasets or imbalanced class distributions. Synthetic data could simulate diverse environmental conditions, plant diseases, and variations, bolstering the model's robustness. Coupled with this, implementing data augmentation techniques during training could further improve generalization by exposing the model to a broader range of transformations, such as rotations, flips, and intensity adjustments.

Third, clustering segmented regions into healthy and PVY-infected categories could provide a more nuanced understanding of disease distribution within a plant. This step would enable fine-grained insights at the regional level, allowing researchers and practitioners to target specific areas of infection more effectively. Additionally, incorporating GPS or location data into the pipeline would enable the precise geolocation of PVY infections within a field, significantly enhancing the application of precision agriculture techniques.

Additionally, we could explore incorporating advanced blur detection mechanisms to identify blurry samples directly, rather than relying on the Laplacian method used in this research. For instance, techniques such as Efficient Image Blur Detection via Hierarchical Edge Guidance and Region Complementation [35] and MSDU-Net: A Multi-Scale Dilated U-Net for Blur Detection [36] offer promising alternatives. Furthermore, future work could analyze the model's performance across varying levels of image quality, particularly on blurry samples. Techniques like heatmaps or GRADCAM could be employed to visualize the relationship between image quality and performance, providing valuable insights into the model's robustness and areas for improvement when handling low-quality samples."

Furthermore, we propose evaluating the use of cosine similarity as an alternative to Euclidean distance for feature comparison. While Euclidean distance was chosen for its ability to capture geometric proximity in the embedding space, cosine similarity, which measures angular relationships, may offer complementary insights. A comparative analysis of these metrics could help further optimize the performance of classifiers in PVY disease detection in potato leaves.

Finally, enhancing plant-level analytics could strengthen the conclusions drawn from the segmentation and classification tasks. Aggregating predictions across multiple leaves or regions could provide comprehensive plant-level statistics, offering robust evidence to determine whether a plant is healthy or diseased. This advancement would add a higher level of confidence to the health assessments, supporting better decision-making processes.

Regarding the overall methodology, integrating these enhancements into the current framework would make it more scalable and impactful. The addition of feature embedding exploration, generalization improvements, synthetic data generation, data augmentation, clustering, geolocation, and plant-level statistics would collectively push the boundaries of precision agriculture, enabling more effective monitoring and management of plant health.

Acknowledgements

This project is financially supported by the European Union within the JTF-call project DigiAgro3.









References

[1] R. E. Goodhue C. L. Carroll, C. A. Carter and C.-Y. C. Lin Lawell. Crop disease and agricultural productivity. Working Paper 23513, National Bureau of Economic Research, June 2017.

[2] C. Leeuwis M. Stuiver and J. D. van der Ploeg. *The Power of Experience: Farmers' Knowledge and Sustainable Innovations in Agriculture*, pages

93–118. European Perspectives on Rural Development. Van Gorcum, 2004.

[3] M. C. Plazas E. E. Vuletic E. Brücher G. Guerra R. De Rossi, F. Guerra and E. M. Reis. Crop damage, economic losses, and the economic damage threshold for northern corn leaf blight. *Crop Protection*, 154:105901, 12 2021.

[4] J. Flood. The importance of plant health to food security. *Food Security*, 2:215–231, 09 2010.

[5] L. Willocquet S. Savary, P. Teng and F. Nutter Jr. Quantification and modeling of crop losses: A review of purposes. *Annual review of phytopathology*, 44:89–112, 02 2006.

[6] Centraal Bureau voor de Statistiek (CBS). Ondernemers zien werkdruk toenemen als gevolg personeelstekort. 2022.

[7] Land en Tuinbouworganisatie Nederland (LTO). Peiling werknemerstekort voor teelt-, productie- en oogstwerkzaamheden. Technical report, 2 2022.

[8] J. E. Wolters R. Voetman, W. Dijkstra and K. Dijkstra. Identification of potato virus y in potato plants using deep learning and gradcam verification. In Kohei Arai, editor, *Intelligent Systems and Applications*, pages 223–244, Cham, 2024. Springer Nature Switzerland.

[9] J. Potgieter M. H. Saleem and K. M. Arif. Plant disease detection and classification by deep learning. *Plants*, 8(11), 2019.

[10] Y. Liu X. Qiao H. Tian, T. Wang and Y. Li. Computer vision technology in agricultural automation —a review. *Information Processing in Agriculture*, 7(1):1–19, 2020.

[11] Z. Mughal M. Awais F. S. Khan M. Saqlain S. Mahamad R. Mahum, H. Munir and I. Tlili. A novel framework for potato leaf disease detection using an efficient deep learning model. *Human and Ecological Risk Assessment: An International Journal*, pages 1–24, 4 2022.

[12] A. Singh S. Mishra and V. Singh. Application of MobileNet-v1 for Potato Plant Disease Detection Using Transfer Learning. *2021 Workshop on Algorithm and Big Data*, 3 2021.

[13] G. Ali S. H. Almotiri M. A. AlGhamdi J. Rashid, I. Khan and K. Masood. Multi-Level Deep Learning Model for Potato Leaf Disease Recognition. *Electronics*, 10(17):2064, 8 2021.

[14] L. M. A. El-Maged N. E. M. Khalifa, M. H. N. Taha and A. E. Hassanien. Artificial Intelligence in Potato Leaf Disease Classification: A Deep Learning Approach. *Studies in Big Data*, pages 63–79, 12 2020.

[15] D. Münch V. Buhrmester and M. Arens. Analysis of Explainers of Black Box Deep Neural Networks for Computer Vision: A Survey. *Machine Learning and Knowledge Extraction*, 3(4):966–989, 12 2021.

[16] A. Maki M. Buda and M. A. Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 106:249–259, 2018.

[17] J. M. Zurada J. Y. Lo-J. A. Baker M. A. Mazurowski, P. A. Habas and G. D. Tourassi. Training neural network classifiers for medical decision making: The effects of imbalanced datasets on classification performance. *Neural Networks*, 21(2):427–436, 2008. Advances in Neural Networks Research: IJCNN '07.

[18] Z. Qi J. Yang, R. Xu and Y. Shi. Visual anomaly detection for images: A systematic survey. *Procedia Computer Science*, 199:471–478, 2022. The 8th International Conference on Information Technology and Quantitative Management (ITQM 2020 2021): Developing Global Digital Economy after COVID-19.

[19] L. R. B. Schomaker K. Dijkstra, J. van de Loosdrecht and M. A. Wiering. Hyper-spectral frequency selection for the classification of vegetation diseases. *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, 2017, 2017.

[20] W. Albattah K. Khan, R. U. Khan and A. M. Qamar. End-to-End Semantic Leaf Segmentation Framework for Plants Disease Classification. *Complexity*, 2022:1–11, 5 2022.

[21] S.-T. Xu Z.-Q. Zhao, P. Zheng and X. Wu. Object detection with deep learning: A review, 2019.

[22] G. Synnaeve-N. Usunier A. Kirillov N. Carion, F. Massa and S. Zagoruyko. End-to-end object detection with transformers, 2020.

[23] N. Parmar-J. Uszkoreit L. Jones A. N. Gomez L. Kaiser A. Vaswani, N. Shazeer and I. Polosukhin. Attention is all you need, 2023.

[24] L. Lu-B. Li X. Wang X. Zhu, W. Su and J. Dai. Deformable detr: Deformable transformers for end-to-end object detection, 2021.

[25] N. Ravi-H. Mao C. Rolland L. Gustafson T. Xiao S. Whitehead A. C. Berg W.-Y. Lo P. Dollár A. Kirillov, E. Mintun and R. Girshick. Segment anything, 2023.

[26] Y.-T. Hu R. Hu C. Ryali T. Ma H. Khedr R. Rädle C. Rolland L. Gustafson E. Mintun J. Pan K. V. Alwala N. Carion C.-Y. Wu R. Girshick P. Dollár N. Ravi, V. Gabeur and C. Feichtenhofer. Sam 2: Segment anything in images and videos, 2024.

[27] H. Zhang F. Li L. Li J. Wang L. Wang J. Gao X. Zou, J. Yang and Y. J. Lee. Segment everything everywhere all at once, 2023.

[28] A. Kolesnikov D. Weissenborn X. Zhai T. Unterthiner M. Dehghani M. Minderer G. Heigold S. Gelly J. Uszkoreit A. Dosovitskiy, L. Beyer and N. Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021.

[29] K. O'Shea and R. Nash. An introduction to convolutional neural networks, 2015.

[30] Poornima Singh Thakur, Pritee Khanna, Tanuja Sheorey, and Aparajita Ojha. Vision transformer for plant disease detection: Plantvit. In Balasubramanian Raman, Subrahmanyam Murala, Ananda Chowdhury, Abhinav Dhall, and Puneet Goyal, editors, *Computer Vision and Image Processing*, pages 501–511, Cham, 2022. Springer International Publishing.

[31] A. Kolesnikov D. Weissenborn X. Zhai T. Unterthiner M. Dehghani M. Minderer G. Heigold S. Gelly J. Uszkoreit A. Dosovitskiy, L. Beyer and N. Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021.

[32] S. Ren K. He, X. Zhang and J. Sun. Deep residual learning for image recognition, 2015.

[33] S. Hochreiter. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 6:107–116, 04 1998.

[34] M. Tan and Q. V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks, 2020.

[35] Xuewei Wang, Xiao Liang, Shaohua Li, and Jinjin Zheng. Efficient image blur detection via hierarchical edge guidance and region complementation. *Complex Intelligent Systems*, 9(6):6523–6540, December 2023.

[36] Xiao X, Yang F, and Sadovnik A. Msdu-net: A multi-scale dilated u-net for blur detection. *Sensors (Basel)*, 21(5):1873, March 2021.

A SOFTWARE & HARDWARE SPECIFICATIONS

Hardware Specifications	
Hardware	**Virtual Machine**
CPU Model	Intel(R) Xeon(R) Gold 6338
CPU	16 Cores @ 2.00GHz
RAM	47 GB
GPU	NNVIDIA A40-12C (Accessed via VMware SVGA II Adapter)
GPU Memory	12 GB
CUDA Version	12.4

Software Specifications	
Software	**Version**
PyTorch	2.5.1+cu124
Torchvision	0.20.1+cu124
Python	3.11.10