IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)
An International Open Access, Peer-reviewed, Refereed Journal

CLASSIFYING LEAF DISEASE SEVERITY USING TRANSFORMER-BASED EXPLAINABLE ARTIFICIAL INTELLIGENCE

[1]Navyashree KS , [2] Ravikiran K V,
[1]Assistant Professor, [2]Data Scientist
[1]Dept of CSE(Data Science), [2]Technology & Research
[1]RNSIT, Bangalore, India, [2]DCIL, Bangalore, India

Abstract: Agribusiness serves as the primary livelihood for approximately 70% of rural residents, with India ranking as the world's second-largest producer of various agricultural commodities. The agriculture industry significantly impacts India's gross domestic product (GDP). Technological advancements play a significant influence in enhancing agricultural practices, enabling the prediction of considerations include soil quality, crop health, and disease detection to optimize crop yields.Detecting and addressing plant leaf diseases early is paramount to prevent their spread and enhance overall yield. This research focuses on plant leaf disease detection and stage classification based on infection severity. The approach involves leveraging deep learning models, specifically YOLOv5 for disease detection, followed by background removal using the U2-Net architecture. Subsequently, a vision transformer (ViT) is employed for stage classification, categorizing severity into low, moderate, and high.To combat leaf diseases effectively, a recommendation solution is proposed. YOLOv5 is trained using open-source datasets, namely PlantDoc and Plantvillage. This study predominantly concentrates on apple leaf disease classification. Results indicate that YOLOv5 achieves a maximum F1-score of 0.57 at a confidence score of 0.2, while the vision transformer, when applied with and without background images, attains F1-scores of 0.758 and 0.908, respectively.

***Index Terms* -** YOLOv5 (You Only Look Once Version 5),ViT (Vision Transformer),CADS (Computer-Aided Disease Detection System),RPN (Region Proposal Network),NLP (Natural Language Processing),XAI (Explainable Artificial Intelligence),DCNN (Deep Convolutional Neural Network).

I. INTRODUCTION

The agricultural sector holds paramount importance in India, serving as the cornerstone of its economy. In this country, farming stands as the predominant occupation and a major contributor to income [1]. India ranks second globally in terms of population, with approximately 70% of its populace residing in rural areas, where agriculture serves as their primary livelihood. Farmers engage in the cultivation of diverse crops, influenced by a multitude of factors, including crop health, environmental conditions, soil quality, local agricultural traditions, the emergence of new pathogen strains, and the presence of various diseases, among 37 others.Farmers are currently grappling with losses due to shifting climatic patterns, with plant diseases posing a substantial threat to food security by significantly diminishing crop quality and yield [2].

The accurate and early diagnosis of diseases has proven challenging. To enhance crop productivity, various methods such as soil management, crop rotation, irrigation, genetic modification, precision farming, weed control, and pest and disease management are employed. Therefore, the timely identification of diseases and appropriate pesticide application is crucial for maximizing crop production. Traditional disease identification methods rely on visual inspection [3], This might result in inappropriate pesticide application and misdiagnosis, having long-term negative impacts on both crops and soil. Hence, there is a growing demand for a Computer-Aided Disease Detection System (CADS) [4] empowered by artificial intelligence (AI) to assist farmers in optimizing their crop yields. CADS incorporates computer vision, machine learning, and a decision support system to detect diseases using visual data.

These CADS Approaches for identifying plant diseases leverage deep learning architectures and are developed on high-computational-resource platforms,, utilizing open-source datasets like PlantDoc [5], PlantVillage [6], aadditionally to specialised plant disease datasets.Additionally, transfer learning approaches are included into CADS systems to help with the diagnosis of illnesses like black rot,bacterial plaque, and rust, utilizing Region Proposal Network (RPN). This method has demonstrated superior accuracy compared to traditional approaches [7], albeit with a more intricate runtime.
Using deep learning models, significant strides in accurately classifying leaf diseases. However, their limited interpretability has hindered their practical adoption. Explainable Artificial Intelligence (XAI) addresses this challenge by shedding light on how AI models make decisions, enhancing their transparency and trustworthiness.Transformer networks, a class of deep learning architecture primarily created for natural language processing (NLP), have demonstrated promising outcomes in a number of computer vision tasks. These networks consist of multiple layers of self-attention mechanisms. Recent advancements in computer vision, which achieve state-of-the-art accuracy while improving parameter efficiency, exemplify how machine learning innovations hold immense potential as a versatile learning technique applicable to diverse data modalities.n essence, XAI and the integration of transformer networks represent two parallel paths in AI research, where the former focuses on interpretability, while the latter showcases the versatility and potential of machine learning across different data types. These developments collectively contribute to the evolution of AI systems for practical applications in various domains.

This paper's primary goal is to leverage deep learning networks to accurately identify and classify plant diseases while providing tailored treatments to prevent the disease from spreading to other plant leaves. Notably, this research aims to address certain shortcomings observed in prior studies, some of which relied on outdated methods or time-consuming features. Considering the literature review, the project is particularly motivated by the need to introduce stage classification, considering the proportion of disease severity, which was previously overlooked. Consequently, this paper places a strong emphasis on this aspect. In this study, we utilize the You Only Look Once version 5 (YOLOv5) deep learning 44 technique for plant detection. Additionally, we utilize a Vision Transformer (ViT) classifier to perform stage classification. The severity of disease on leaves is categorized into three stages: low severity (0 to 30%), moderate severity (31 to 60%), and high severity (61 to 100%).

The structure of the paper is outlined as follows:

Section 1:provides an introduction to the agricultural context, highlights the limitations of previous literature, outlines the research's motivation, objectives, and contributions.

Section 2:provides an extensive literature review on the identification and classification of plant diseases.

Section 3:delves into the various methods employed for grading of leaf disease severity. Additionally, it covers the multistage leaf disease classification method and the experimental setting used to train models for leaf disease detection.

Section 4:gives the findings, including assessments of leaf performance. Section 4 presents the results, including performance evaluations of detection of leaf disease using the PlantDoc dataset, analysis of training and testing, assessment of background removal techniques, multistage leaf disease classification, and specific results for apple rust and apple scab. This section culminates with a comparative analysis of multi-stage leaf disease classification, with and without background, employing Vision Transformer (ViT).

In summary, this paper integrates state-of-the-art deep learning techniques to enhance the detection of plant diseases and presents a novel method for stage categorization based on disease severity. The comprehensive The paper's structure guarantees clarity in presenting the research's objectives, methodologies, and findings.

II. Literature review

Numerous important elements, including as changes Climatic factors such as 45 temperature, humidity, soil moisture, light intensity, and pH levels have an influence on leaf diseases.
The health of plants is also greatly influenced by harmful organisms like viruses, bacteria, fungus, and insects. Several researchers have explored various approaches to 47 mitigate leaf diseases.

Dhaygude and Kumbhar [10] introduced the application of textural statistics for diagnosing plant leaf diseases. Their approach involved transforming HSV colour from RGB photos, removing green pixels based on threshold values, segmenting images into 32x32 patches, and calculating texture statistics using color occurrence techniques to assess disease presence. However, their literature review lacks details on specific vision-based algorithms and neural networks that could enhance classification, as well as potential difficulties in putting them into practice.

Sethy et al. [11] employed K-means clustering, Multiclass-SVM, and particle swarm 29 optimization (PSO) to detect rice leaf diseases. They employed the gray-level cooccurrence matrix (GLCM) for feature extraction, and the utilization of Particle Swarm Optimization (PSO) led to a substantial improvement in detection accuracy, achieving a rate of 97.91%.However, the study's generalizability is limited due to a small dataset with restricted variations of infected rice leaves.

Khirade and Patil [13] suggested the application of image processing and the Internet of Things (IoT) for the detection of plant diseases, with a specific focus on hill 1 banana leaves.They captured images with a camera sensor and transmitted them to cloud storage using the Raspberry Pi3 model. The study utilized a random forest classifier with GLCM features to detect diseases like Black Sigatoka, Bunchy Top virus, and Yellow Sigatoka. However, the study lacks performance evaluation on the dataset and is limited to the hill banana dataset.

Cao et al. [14] suggested a multi-scaled convolutional object detection network that leveraged deep convolutional networks and deformable convolutional structures to handle geometric alterations. Their framework improved accuracy in recognizing small target objects with geometric deformations. However, its applicability is somewhat constrained, primarily focusing on video object detection networks.

Guo et al. [15] suggested a deep learning-based mathematical model for detecting and recognising plant diseases.They used a Region Proposal Network (RPN) and the Chan-Vese algorithm to locate leaves and segment images based on symptoms. The model outperformed conventional methods in detecting rust, bacterial plaque, and black rot diseases with an accuracy of 83.57%. However, the Chan-Vese algorithm's iterative calculation time could be optimized to expedite the process. Future research could consider constructing zero initial sets using neural networks to facilitate quicker identification results. This algorithm holds significant promise for ecological preservation, sustainable agriculture, and intelligent farming, despite The necessity for... address computational efficiency considerations related with the Chan-Vese technique.

Hassan et al. [17] investigated the use of deep convolutional neural network (DCNN) models for the identification and diagnosis of plant diseases using leaf imagery. To limit the number of parameters and processing expenses, these models were created 60 58 using depth separable convolution. They were taught using a dataset encompassing 14 plant species and 38 different disease classes. The results demonstrated impressive disease classification accuracy rates, surpassing custom-feature-based methods, and requiring less training time compared to previous deep learning models. The study also highlighted the potential for implementing these models on mobile devices using the MobileNetV2 architecture. In conclusion, DCNN models show promise for efficient and real-time identification of plant diseases in agricultural systems.

Rashid et al. [18] addressed the the difficulties associated with early detection of 18 potato leaf diseases (PLD), considering variations in crop types, disease symptoms, and environmental factors. While currently available machine learning approaches 25 for PLD identification are location-specific, the paper proposed a multi-level early deep learning model blight and late blight disease identification in potato leaves. This model, incorporating image segmentation and a convolutional neural network, achieved an impressive accuracy rate of 99.75% on a dataset collected from the Central Punjab region of Pakistan. However, a limitation is that it focuses solely on detecting a specific disease on a leaf, without assessing disease location or severity.

Additionally, the PLD dataset could be enhanced, and future research should consider the creation of an Internet of Things-based monitoring system, website, and mobile application.

YOLOv1, introduced by Redmon et al. [19], iis founded on Darknet framework and trained on the ImageNet-1000 dataset with a 224x224 input picture size. It boasts a rapid object identification capability at 45 frames per second. However, it struggles with object generalization when images have varying dimensions and may not accurately identify tiny objects.

YOLOv2 and YOLO9000, as proposed by Redmon and Farhadi [20], adopt the Darknet 19 architecture comprising 19 convolutional layers, 5 max-pooling layers, and a SoftMax layer. This architecture demonstrates advancements in object detection capabilities.

YOLOv3, introduced by Redmon and Farhadi [21], employs the Darknet-53 network, which consists of 53 convolutional layers, as its feature generator. With an input image size of 320×320, it primarily consists of 3x3 and 1x1 filters with shortcut links and uses feature pyramid network (FPN) to make class forecasts. More than 80 distinct items can be recognized by YOLOv3 in a single image.

YOLOv4, as proposed by Bochkovskiy et al. [22], employs cross-stage partial architecture (CSPDarknet53), a bag of freebies, and a bag of specials as architectural styles. Features like cross-iteration batch normalization (CBN), pan aggregation network (PAN), weighted-residual-connections (WRC), and cross-stage-partial connections (CSP) are applicable to the majority of tasks, models, and datasets. YOLOv4 was rated as one of the best models for speed and accuracy for the Common Objects in Context (COCO) dataset, even if its overall accuracy lagged behind that of EfficientDet largest model. However, YOLOv4 struggles to recognize because each grid has close objects can suggest only two bounding boxes.

YOLOv5, created by Ultralytics open-source research on cutting-edge vision AI techniques [23], uses a 416x416 input image size and features like auto-learning bounding box anchors, 16-bit floating point precision, and new model configuration files.It offers precise and efficient object detection through the use of .yaml files, cross-stage partial networks as the backbone, preliminary assessment metrics, and other innovations.

"Faster R-CNN, as explored by Ren et al. in their research [24], is centered around achieving real-time object detection through the utilization of Region Proposal Networks (RPN)." It leverages fully trained region proposals from RPN to make accurate detections and combines RPN and Faster R-CNN into a A single network containing convolutional features that are shared.. This approach achieves state-of-the-art object detection accuracy on datasets like PASCAL VOC2007, VOC2012, and MS COCO with just 300 proposals per image.

Tan and Le [28] introduced a novel scaling method that equally scales depth, breadth, and resolution parameters using a compound coefficient. This method, known as EfficientNets, outperformed previous ConvNets iin terms of precision and efficiency. EfficientNet-B7, for example, achieved better performance while being significantly faster and smaller at inference.

"In their study, Malik and colleagues [29] detailed the application of deep learning models like VGG-19, ResNet-34, and ResNet-50 for the detection of diseases in sugarcane crops." "In their research, the study concentrated on the detection of five distinct sugarcane diseases, making use of images captured under varying resolutions and illumination conditions. Their proposed model demonstrated resilience in recognizing intricate patterns, achieving an accuracy rate of 93.20% on the testing dataset and 76.40% on data obtained from online sources. Nonetheless, the study acknowledged the necessity for a larger dataset to enhance overall results. In a separate study, Oda and Okura [30] explored different visualization techniques at both the layer-wise and neuron-wise levels. They trained neural networks using publicly accessible datasets, showcasing the capacity of these networks to capture unique texture and color characteristics associated with diseases. They streamlined their model by removing non-contributory layers, reducing parameters by 75% without compromising accuracy. However, their study was constrained by limited training data, which could potentially lead to overfitting, and lacked transparency in explaining the decision-making process of deep neural networks."

"In their study, Nagasubramanian and colleagues [31] presented an inventive method that harnesses a 3D deep convolutional neural network (DCNN) in conjunction with hyperspectral data for identifying charcoal rot in soybean crops. The model achieved an impressive F1 score of 0.87 for the infected class and exhibited remarkable accuracy, with a classification accuracy rate of 95.73%. Despite the relatively limited dataset, the model's robustness was extensively validated through a rigorous fivefold cross-validation procedure."

"In an independent study, Liu and Wang [32] outlined a technique for the early detection of tomato leaf spots using the MobileNetv2-YOLOv3 model, incorporating a GIoU bounding box regression loss function. Their technology demonstrated improved detection accuracy and real-time capabilities, achieving a high F1 score of 94.13%, a substantial likelihood of 92.53%, and an impressive Intersection over Union (IOU) value of 89.92%. This innovative approach holds significant potential for practical implementation in crop farming through computer or mobile applications."

"In their study, Malik and his team [33] presented a hybrid deep learning model that combines MobileNet and VGG-16 models through stacked ensemble learning for the purpose of sunflower disease classification. Despite the challenges posed by distinct sunflower species and similar disease symptoms, their model achieved an accuracy rate of 89.2%. Nevertheless, they acknowledged the necessity for a more extensive dataset to enhance the quality of results."

"In a study conducted by Tammina [34], the application of transfer learning with VGG-16 for image recognition was discussed. Through image augmentation, the model's accuracy increased from 72.40% to 79.20%, ultimately reaching an impressive accuracy of 95.40% when using pre-trained weights.In a separate study, Nagasubramanian et al. [31] introduced an innovative approach involving a 3D deep convolutional neural network (DCNN) that incorporated hyperspectral data for the detection of charcoal rot in soybean crops. This model achieved an F1 score of 0.87 for the infected class and a classification accuracy of 95.73%. Furthermore, it provided valuable physiological insights into its predictions. Despite the relatively smaller dataset size, the model's robustness was rigorously verified through a fivefold cross-validation process."

"In a study by Liu and Wang [32], they employed the MobileNetv2-YOLOv3 model with a GIoU bounding box regression loss function for the early detection of tomato leaf spots. Their technology not only demonstrated improved detection accuracy but also real-time capabilities, boasting a high F1 score of 94.13%, a substantial likelihood of 92.53%, and an impressive Intersection over Union (IOU) value of 89.92%. The proposed method holds significant potential for practical applications in crop farming, including computer and mobile applications."

"In their study, Malik and colleagues [33] presented a novel hybrid deep learning model that incorporated both MobileNet and VGG-16 models through stacked ensemble learning for the purpose of classifying sunflower diseases. Despite encountering challenges associated with distinct sunflower species and similar disease symptoms, their model achieved a noteworthy accuracy rate of 89.2%. Nevertheless, they acknowledged the importance of a more extensive dataset to further enhance the quality of results."

"In a separate investigation, Tammina [34] utilized transfer learning with VGG-16 for the purpose of image recognition. Remarkably, the model's accuracy exhibited a significant improvement, increasing from 72.40% to 79.20% when image-level features were incorporated. This methodology not only achieved high accuracy but also yielded valuable insights into physiological predictions. Despite the relatively limited dataset size, the model's reliability and robustness were rigorously validated through a fivefold cross-validation process."

"In the study conducted by Liu and Wang [32], they presented a method for the early detection of tomato leaf spots using the MobileNetv2-YOLOv3 model along with a GIoU bounding box regression loss function. Their system showcased substantial improvements in detection accuracy and real-time capabilities, achieving impressive F1 scores of 94.13%, a high probability rate of 92.53%, and an average Intersection over Union (IOU) value of 89.92%. This proposed algorithm holds significant promise for practical implementation in crop farming and could facilitate its adoption through computer or mobile applications."

Malik et al. [33] described aa deep learning hybrid model that incorporates MobileNet and VGG-16 models using stacked ensemble learning for sunflower disease classification. Despite challenges posed by different sunflower species and similar disease symptoms, their The model attained a precision of 89.2%. However, they acknowledged the need for a larger dataset to improve results.

Tammina [34] explained ted VGG-16 model fine-tuned with a substantial image dataset. Sandler et al. [35] introduced MobileNetV2, a mobile model architecture that excels in various performance metrics and benchmarks. They also presented Single Shot Detector Lite (SSDLite), a real-time object detection framework with improved accuracy and reduced complexity compared to YOLOv2. Their solution outperforms real-time detectors on the COCO dataset while being computationally efficient. Srinidhi et al. [36] successfully employed DenseNet and EfficientNet models to detect four types of apple plant diseases from leaf images. Their models achieved high accuracy (99.8% for EfficientNetB7 and 99.75% for DenseNet) after enhancing the dataset with data augmentation and image annotation techniques.

Sibiya and Sumbwanyambe [37] highlighted the advantages of fuzzy logic principles in the context of the Leaf Doctor application, which assesses the severity of plant leaf 61 diseases. They proposed an updated algorithm that could enhance precision agriculture and be integrated into smartphone apps like Leaf Doctor. This approach simplifies disease severity estimation for non-experts and requires less time than other methods. The recommended color threshold segmentation approach and fuzzy logic inference system were tested, and the study suggests potential applications in future upgrades of the Leaf Doctor app.

Shi et al. [39] conducted a comprehensive review of studies related to plant disease severity assessment using convolutional neural networks (CNN). They categorized these studies into classical CNN frameworks, improved CNN architectures, and CNN-based segmentation networks based on network design. The review provided an in-depth comparative analysis of the sthe pros and weaknesses of each strategy. Additionally, the paper explored common techniques for dataset acquisition and performance evaluation metrics for CNN models. Overall, this literature survey offers valuable insights into the current landscape of deep learning applications in plant disease severity assessment and identifies potential avenues for future research.

Abd et al. [40] introduced a novel deep learning approach called ant colony optimization with convolutional neural network (ACO-CNN) for disease detection and classification in plant leaves. This technique leverages ant colony optimization to enhance disease diagnosis accuracy. The method utilizes a CNN classifier to extract color, texture, and leaf arrangement features from images. The study compared the proposed ACO-CNN approach with existing methods using various effectiveness metrics, demonstrating its superior accuracy. The paper outlined the key steps involved in disease detection, including image acquisition, image separation, noise reduction, and classification. However, it should be noted that the paper does not 53 address the stage classification of plant diseases.

Mohammed and Yusoff [41] presented a comprehensive overview of techniques in the fields of machine learning, deep learning, and image processing, emphasizing the significance of training models with larger datasets to improve recognition accuracy. The review underscored the need for advanced deep learning methods for plant disease classification. CNNs were highlighted as particularly effective due to their ability to extract picture characteristics, surpassing traditional models like naive Bayes and support vector machines (SVMs). CNNs were recognized as the preferred choice for image processing, machine learning, and deep learning research, given their feature extraction capabilities.

The above literature reviews collectively illustrate the progress in plant disease diagnosis using image processing, machine learning, and deep learning techniques. However, many studies have primarily focused on disease detection or classification, with limited attention to stage identification, as a result of which lack of a comprehensive solution for deployment. Consequently, there is a demand for a robust and novel pipeline for plant disease stage classification. This research addresses this need by developing an end-to-end deployable solution that integrates leaf detection using YOLOv5 architecture and stage classification using the Vision Transformer (ViT) architecture [42].

The system also includes a recommendation engine that provides mitigation suggestions based on disease severity. Ground truth for stage classification was established with the assistance of an expert plant pathologist. To streamline the process and reduce the workload, an unsupervised clustering-based methodology was devised for generating ground truth data. Additionally, a comparative analysis of F1 scores was conducted to evaluate multistage leaf disease classification, both with and without background information, using the ViT classifier. Interpretability of the disease stage classification was also explored, focusing on apple leaf diseases.

III. Methods

The end-to-end multistage leaf disease detection and classification system is 3 illustrated in Figure 1, comprising six distinct stages: **1. Data Acquisition**: The system begins by gathering data from the PlantDoc and Plantvillage datasets.

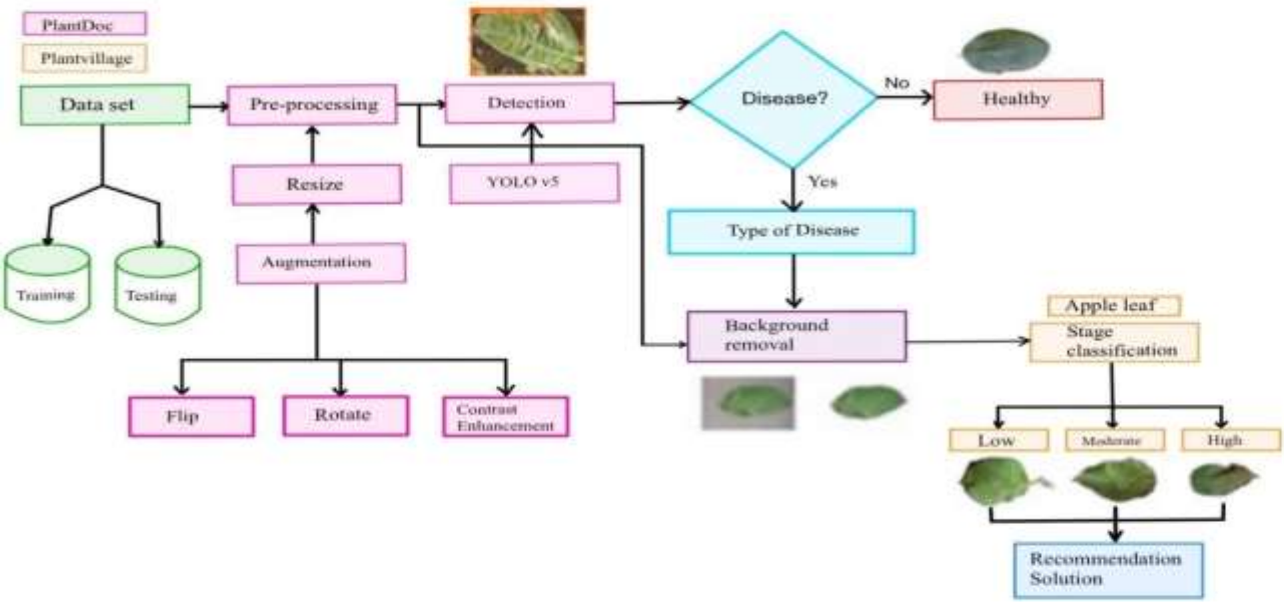
2. Data Preprocessing: The acquired data undergoes preprocessing, which involves operations such as resizing, augmentation, and contrast enhancement.

3. Disease Detection Model Training: A disease detection model is trained with various parameters and hyperparameter tuning to enhance its performance.

4. Background Removal: To achieve accurate image classification, background removal is executed, isolating the leaf and its disease.

5. Stage Classification: "The system categorizes the disease by assessing the severity of leaf infection and distinguishing among various stages, including low, moderate, and high levels of infection."

6. Recommendation: In the final stage, the system provides recommendations for mitigating the identified leaf disease. It's important to note that in this research, apple leaf data was specifically employed for multistage classification. The ground truth for various disease stages (low, moderate, and high) was created through the mean-shift clustering technique and subsequently validated by domain experts.



"Figure 1: illustrates the concept of multi-stage detection and classification of plant leaf diseases."

"In this research, two publicly accessible datasets were employed for the detection and classification of plant leaf diseases:" PlantDoc Dataset: "The PlantDoc dataset [4] has emerged as a valuable asset for developing standardized image classification models. "This dataset contains cropped leaf images and has seen extensive usage among researchers for training various image classification models, including VGG-16, InceptionV3, as well as object identification models like MobileNet and Faster-RCNN." "It plays a crucial role in augmenting the capabilities of computer vision tasks in agriculture, encompassing tasks like crop health assessment and plant disease classification."

The PlantDoc dataset comprises a total of 2,572 images, encompassing 13 different plant species and 27 distinct classes. "Among these categories, 17 classes correspond to diseased leaves, while the remaining 10 classes are associated with healthy leaves. The details of these specific classes and the number of images in each category can be found in Table 1. These images served as the training dataset for training the YOLOv5 model, which was employed for both leaf region detection and disease classification." "It is imperative to provide appropriate attribution by citing the source (reference [4]) whenever utilizing data or information from external sources such as the PlantDoc dataset in your research. Figure 2 illustrates two example classes of apple leaf diseases, namely, apple rust leaf and apple scab leaf."

Figure 2 : "Apple leaf diseases sourced from the PlantDoc dataset."

Table 1: PlantDoc dataset used for leaf disease detection

Classes	Ranges/No. of Images
Tomato two spotted spider mites leaf(leaflet)	2
Tomato mosaic virus leaf(leaflet)	54
Cherry leaf(leaflet)	57
Bell pepper leaf(leaflet)	61
Tomato leaf(leaflet)	63
Grape leaf(leaflet) black rot	64
Soyabean leaf(leaflet)	65
Corn Gray leaf(leaflet) spot	68
Grape leaf(leaflet)	69
Bell pepper leaf(leaflet) spot	71
Tomato yellow virus leaf(leaflet)	76
Apple rust leaf(leaflet)	88
Tomato early blight leaf(leafleApple Leaf(leaflet)	88
Tomato leaf(leaflet) mold	91
Apple Scab Leaf(leaflet)	91
Strawberry leaf(leaflet)	93
Potato leaf(leaflet) late blight	96
Tomato bacterial spot leaf(leaflet)	105
Peach leaf(leaflet)	110
Tomato late blight leaf(leaflet)	111
Blueberry leaf(leaflet)	115
Corn rust leaf(leaflet)	116
Potato leaf(leaflet) early blight	119
Raspberry leaf(leaflet)	119
Squash Powdery mildew leaf(leaflet)	130

Tomato septoria leaf(leaflet) spot	151
Corn leaf(leaflet) blight	191
Healthy	847
Unhealthy	1725
Total	**2572**

The PlantVillage dataset [5] is a comprehensive collection of meticulously labeled images encompassing both healthy and diseased plant leaves belonging to 14 different crop types. These images mark the commencement of an ongoing crowdsourcing initiative aimed at harnessing computer vision techniques to address the issue of crop yield losses stemming from viral diseases. This dataset has achieved great appeal and is widely used by several plant disease detection system researchers.

"In this research, a portion of the PlantVillage dataset was employed for stage classification employing Vision Transformer (ViT) models. This dataset encompasses typical images of apple rust and apple scab, which are depicted in Figure 3." "In preparation for model training, the images were resized to dimensions of 416x416, which is a requirement for the YOLOv5 architecture [43]. To promote better generalization and reduce the potential for overfitting, several data augmentation techniques were employed, including (i) image flipping, (ii) rotation, and (iii) contrast enhancement."
"During the YOLOv5 model training process, the dataset was divided into two distinct segments: 90% of the data was allocated for model training, while the remaining 10% was set aside for assessing the model's performance. A comprehensive breakdown of the distribution of images assigned to training and testing within each disease class in the PlantDoc dataset can be found in Table 2."

Figure 3: Apple leaf diseases from the plantvillage dataset

Table 2: Training and testing data for training leaf disease detection using plantDoc dataset

Class name	"Images used for training."	" Images used for testing"
Tomato two spotted spider mites leaf(leaflet)	2	0
Tomato mosaic virus leaf(leaflet)	44	10
Cherry leaf(leaflet)	47	10
Bell pepper leaf(leaflet)	53	8
Tomato leaf(leaflet)	55	8
Grape leaf(leaflet) black rot	56	8
Grape leaf(leaflet)	57	13
Soyabean leaf(leaflet)	37	6
Blueberry leaf(leaflet)	62	10
Corn rust leaf(leaflet)	64	9
Potato leaf(leaflet) early blight	63	9

Raspberry leaf(leaflet)	78	11
Squash Powdery mildew leaf(leaflet)	101	8
Tomato septoria leaf(leaflet) spot	102	7
Corn leaf(leaflet) blight	179	11
Healthy	757	90
Unhealthy	1579	146
Total	3336	364

3.3 Leaf disease detection :

Leaf disease detection is achieved through the utilization of the advanced object identification technique known as YOLOv5. YOLOv5 employs a single neural network that processes an entire image and subsequently segments it into distinct components while providing bounding box probabilities for each component. "When an image is fed into the network, it generates a set of bounding boxes and corresponding class probabilities as its output. These bounding boxes precisely delineate the object locations within the image, and the class probabilities indicate the likelihood of each object belonging to a particular category."One of the primary enhancements in YOLOv5 is its scalable architecture, which dynamically adjusts the network's size depending on the input image's dimensions.

This adaptability enables YOLOv5 to offer improved accuracy and faster performance compared to its predecessors.

"When considering the training of the YOLOv5 model, two crucial aspects come into play:"

a) **Activation and Optimization Functions**:

- Leaky Rectified Linear Unit (ReLU) and Sigmoid functions are employed as activation functions.
- Stochastic Gradient Descent (SGD) and Adaptive Moment Estimation (ADAM) are used as optimizer functions.
- In YOLOv5, the final detection layer utilizes the sigmoid activation function, while the intermediate hidden layers utilize the Leaky ReLU activation function.

b) **Loss Function:**

- YOLOv5 calculates a compound loss that incorporates the objectness score, class probability score, and bounding box regression score.
- For computing the loss associated with class probability and object score, Ultralytics utilizes the Binary Cross-Entropy with Logits Loss function from the PyTorch library.

The loss function of YOLO is formally described by Equation 1. [Note: It's important to reference any sources or specific implementations (e.g., the PyTorch library) when discussing techniques and functions used in your research.]

$$loss = l_{box} + l_{cls} + l_{obj} \quad (1)$$

where lbox is a bounding box regression function, lcls is a loss function of classification and lobj is a loss function of confidence. The regression loss function of the bounding box is written as in Equation 2.

$$l_{box} = \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{i,j}^{obj} b_j (2 - w_i X h_i) [(x_i - \hat{x}_i^j)^2 + (y_i - \hat{y}_i^j)^2 + (w_i - \hat{w}_i^j)^2 (h_i - \hat{h}_i^j)^2] \quad (2)$$

The classification loss function is written as in Equation 3.

$$l_{cls}=\lambda_{class}\sum_{i=0}^{S^2}\sum_{j=0}^{B} I_{i,j}^{obj}\sum_{c\in classes} p(c)\log(\hat{p}i(c)) \quad (3)$$

The confidence loss function is written as in Equation 4.

$$l_{obj}=\lambda_{noobj}\sum_{i=0}^{S^2}\sum_{j=0}^{B} I_{i,j}^{noobj}(c_i-\hat{c}_j)^2+\lambda_{obj}\sum_{i=0}^{S^2}\sum_{j=0}^{B} I_{i,j}^{obj}(c_i-\hat{c}_j)^2 \quad (4)$$

In the context of YOLOv5, the loss calculation involves several components. These include ˆˆ representing the actual central coordinate of the target, ˆˆ representing the target's width and height, and the introduction of λcoord as the position loss coefficient and λclass as the category loss coefficient.

When assessing whether the anchor box at a particular position (i, j) contains targets (obj), the variable Ii,j takes on a value of 1 if targets are present, and 0 otherwise. Pi(c) denotes the target's category probability, while ˆ (c) represents the actual value of the category. The total number of categories, denoted as C, is determined by the length of these two vectors.

The YOLOv5 model is then tasked with processing the input image and generating an output image that includes detected objects, each accompanied by a confidence score, as depicted in Figure 4.

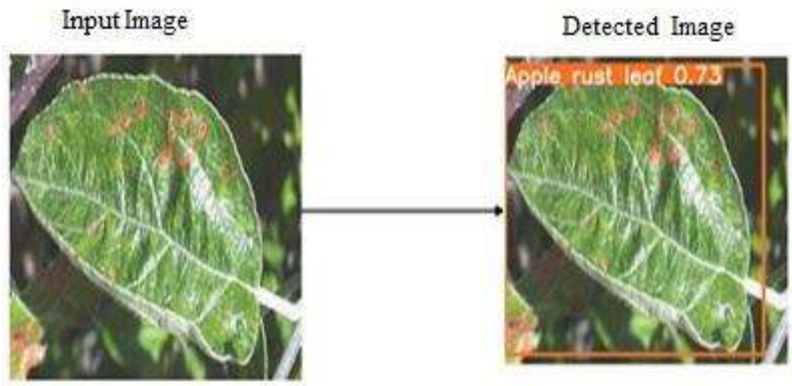


Figure 4:Leaf disease detection using YOLOv5

The following algorithm, Algorithm 1, outlines the procedure for preparing multistage infection ground truth in the context of apple leaf disease images from the PlantVillage dataset. The output of this algorithm is the identification of three distinct severity stages for the apple leaf disease.

Algorithm 1: Multi-Stage Infection Ground Truth Preparation Algorithm

Input: Apple Leaf Disease Images from PlantVillage Dataset Output: Identification of Different Severity Stages as Ground Truth for Apple Leaf Disease Dataset Classification

Step 1: Apply the mean-shift clustering technique to identify clusters representing disease regions [44].

Step 2: "Apply the Canny edge detection method to detect the edges of both the leaf and the infected regions [45]."

Step 3: "Locate and compute the contour of the leaf to ascertain the extent of diseaseaffected areas on the leaf."

Step 4: Calculate various metrics, including leaf infection area, total area, perimeter, and the percentage of infection on the leaf. Step 5: Group the images into three severity categories based on the results obtained 41 in Step 4: (i) low severity, (ii) moderate severity, and (iii) high severity.

"This algorithm enables the generation of ground truth data, which is instrumental in the classification of apple leaf disease images into various severity stages, assisting in the subsequent classification task." "Background removal stands as a crucial digital image processing method employed to separate image components into desired and undesired regions. This technique holds importance across a range of image processing and computer vision applications, ensuring accurate analysis and subsequent processing. To minimize the risk of incorrect classification results, the U2-Net architecture has been selected for background removal [46].

"The U2-Net architecture is structured into two levels of nesting, comprising a total of 11 stages at the top level. Each stage is configured with a residual U-block (RSU) at its base, as illustrated in Figure 5(a). In the specific context of processing apple leaf images, the U2-Net architecture is employed for background removal. This process involves the transformation of the RGB image of the apple leaf into a binary image. Subsequently, the binary image is subjected to processing by the U2-Net model to eliminate the background, with a specific emphasis on the utilization of residual Ublocks. Figure 5(b) visually demonstrates the removal of unwanted background noise from the leaf image—a critical step in ensuring the accuracy of image classification, as extraneous background elements can have adverse effects on classification outcomes." "The process begins with the input of an image into the U2-Net model, which transforms it into a binary image. Subsequently, the U2-Net model skillfully removes background noise from this binary image. Finally, the model proceeds to eliminate background noise from the original RGB image, resulting in the desired output image."

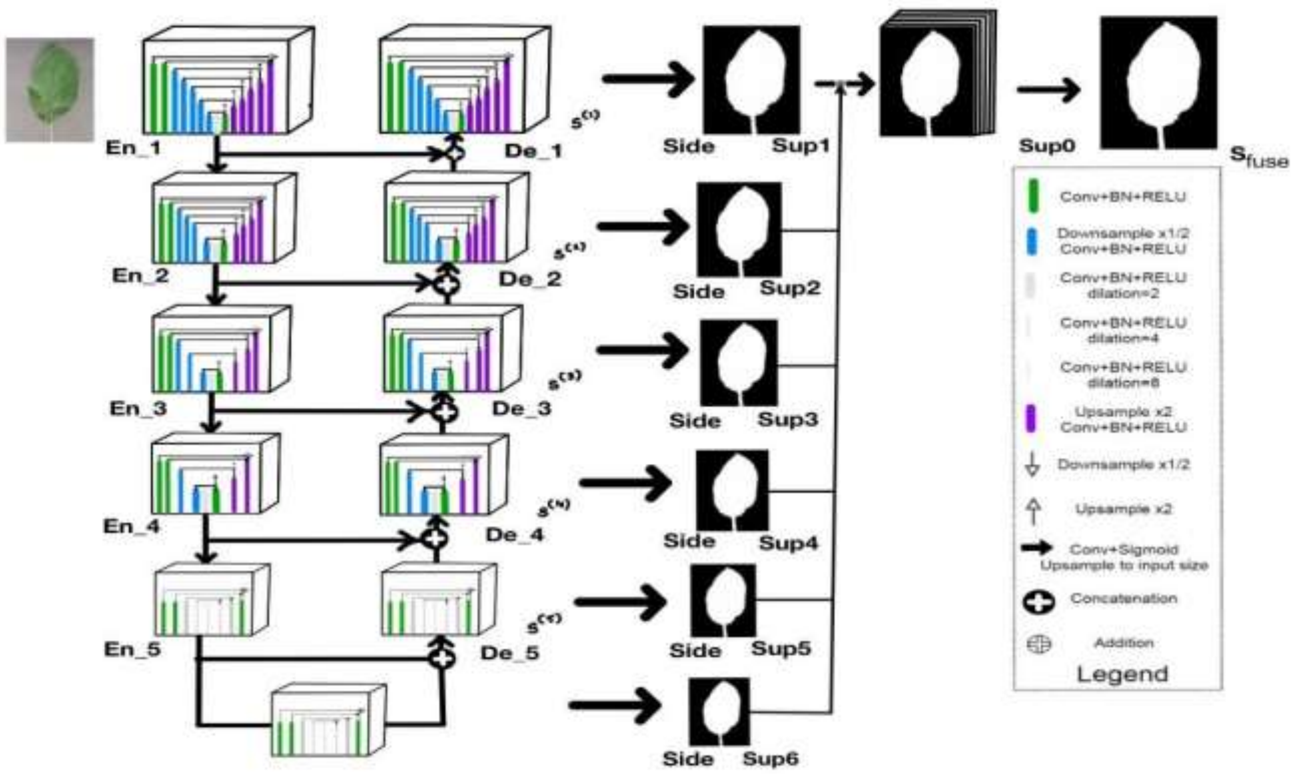


Figure 5(a) :Illustration 0f U2-Net architecture

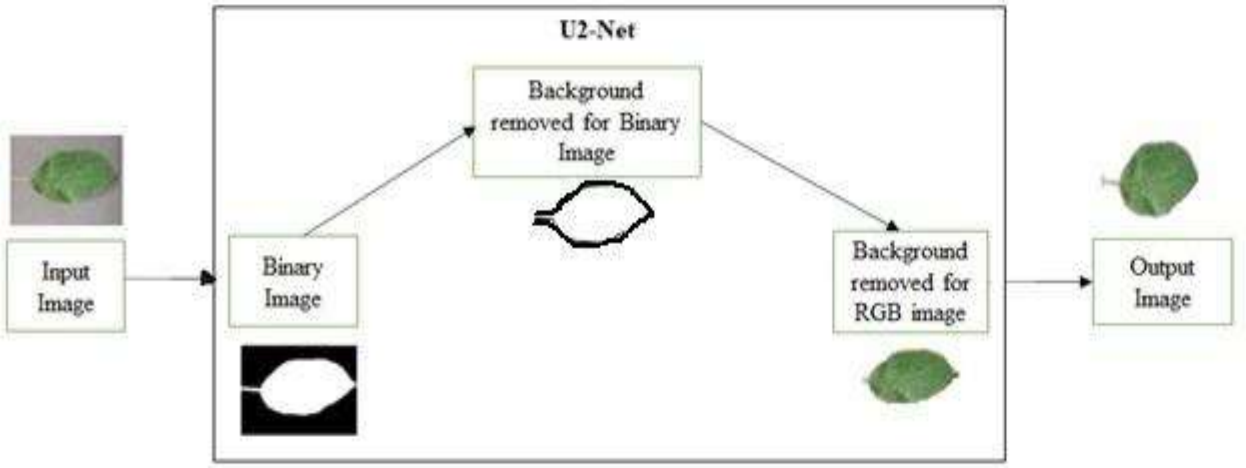


Figure 5(b) :Plant leaf image background removal using U2-Net architecture

The proposed multistage classification approach harnesses the power of the ViT (Vision Transformer) deep learning architecture. ViT is a technique primarily designed for image classification, utilizing a transformer-like architecture that divides images into patches. "These patches, each with dimensions of 16x16 pixels, are converted into flattened vectors representing pixel values and linearly embedded. Subsequently, these embedded patches are aggregated and processed through a conventional transformer encoder, as depicted in Figure 6. Vision Transformer (ViT) is an amalgamation of techniques from natural language processing (NLP) and computer vision, showcasing outstanding performance in diverse computer vision applications. This includes object detection, image classification, computer vision tasks, and semantic segmentation. ViT applies transformers directly to sequentially organized image patches, rendering it highly effective for image classification assignments."

Algorithm 2: Inference Leaf Disease Detection and Stage Classification Input: Plant Leaf Image I(x, y) from PlantDoc Dataset Output: Detected Disease ID(x, y), Severity of Disease SID(x, y)

- PD → Plant Disease Type
- IROI(X, Y) → Region of Interest of Diseased Image
- BR() → Background Removed Image
- ViT() → Vision Transformer
- ROI BR(x, y) → Region of Interest of Background Removed Image

Step 1: Feed the image I(x, y) into the YOLOv5 disease detection model for disease detection.

Step 2: "Examine the image to assess its state of health, making a distinction between healthy and diseased conditions."

Step 3: "If the image is detected as diseased, locate the Region of Interest (ROI) within the diseased image, designated as

IROI(X, Y)."

Step 4: Perform background removal on IROI(X, Y) to obtain a backgroundremoved image, BR().

Step 5: Apply a Vision Transformer (ViT) to the ROI of the background-removed image to determine the severity of the disease, resulting in SID(x, y). This algorithm outlines the steps involved in disease detection and stage classification for plant leaf images, utilizing YOLOv5 for initial detection and ViT for disease severity assessment.

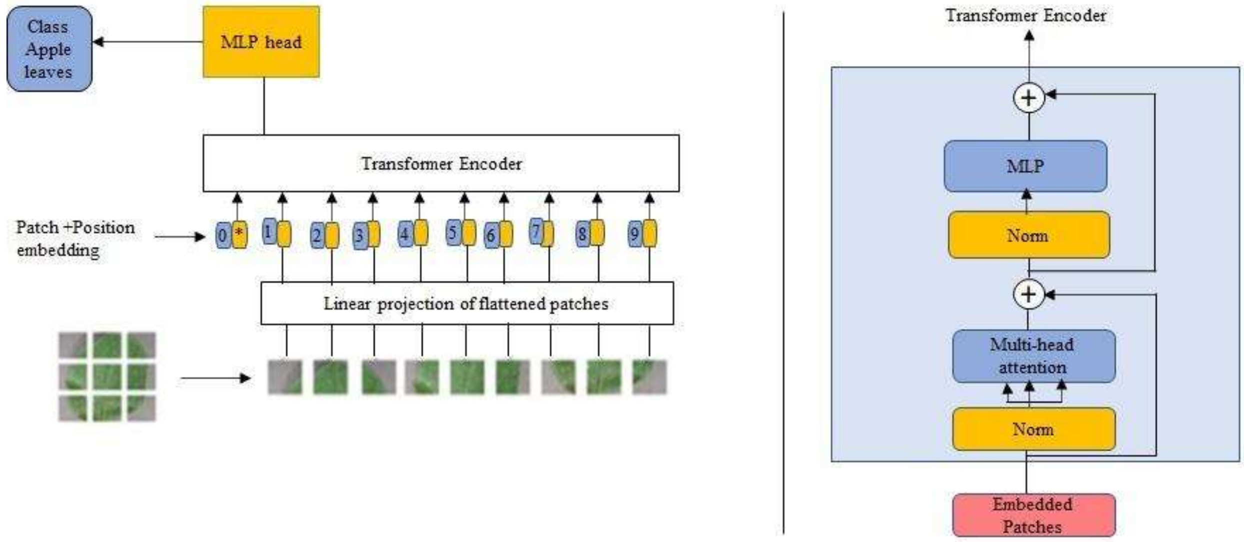


Figure 6 :Multistage classification using vision transformer "

The experimental setup for this research project was carried out on the Google Colab platform, which is an open-source platform, utilizing a K80 NVIDIA GPU." The software environment consisted of the PyTorch library combined with Python 3.7. In terms of hardware, the requirements included an Intel Core i7-12th generation system, a 1TB hard disk, and 32GB of RAM. The software resources comprised the Windows 10 operating system, Google Colab with K80 NVIDIA GPU as the software tool, Python 3.7 as the coding language, and the PyTorch library.

Training Parameters and Hyperparameters (Table 3): For training the YOLOv5 model using the PlantDoc dataset, the following parameters and hyperparameters were employed:

- Input image size: 416x416 dimensions
- Batch size: 50 ("indicating the volume of training data processed in a single iteration") 8
- Number of epochs: 715
- Learning rates: 0.1 and 0.01
- Weight decay: 0.0005

Table 3 Training parameters/hyperparameters for leaf disease detection using YOLOv5

S. No	Training parameters		Values
1	Image size		416 × 416
2	Batch size		50
3	Epochs		715
4	Learning rate	lr 0	0.01
		lr f	0.1
5	Weight decay		0.0005

For the multistage classification of plant leaf disease severity using the Plant Village dataset, we have utilized specific training parameters, as outlined in Table 4:

Training Parameters for Multistage Classification (Table 4):

- Input image size: 224x224 dimensions

- Batch size: 64
- Number of epochs: 200
- Learning rate: 2e-5
- Gamma: 0.7 (utilized to monitor training progress)
- Seed value: 42

The selected disease class for this classification task is apple leaf disease, and it involves both training and testing data using the ViT (Vision Transformer) classifier. Images are considered with both the background and without background removal.

"Data Statistics and Data Augmentation (Tables 5 and 6): Table 5 provides an overview of the dataset distribution utilized for training, testing, and validation, including the count of images assigned to each stage of the ViT classifier. Ensuring an equitable distribution of images for training, testing, and validation is of paramount importance." To mitigate potential data imbalance issues, we applied data augmentation techniques. These techniques included rotation, flipping, and rotation at various angles, all aimed at achieving an equitable distribution of images for each stage of the classification. The training parameters and data augmentation strategies are designed to improve the ViT classifier's effectiveness in precisely categorizing the severity stages of apple scab and rust leaf diseases using the Plant Village dataset.

Table 4 Training parameters/ hyperparameters for leaf disease classification using ViT

S. No.	Training parameters	Values
1	Image size	224 x 224
2	Batch size	64
3	Epochs	200
4	Learning rate	2e-5
5	Gamma	0.7
6	Seed	42

Table 5 Training, testing, and validation data for multistage leaf disease classification using plantvillage dataset with background images

Type of disease class	Stage of the disease	Train	Test	Validation
Apple Scab Leaf	Apple Scab Low	217	217	217
	Apple Scab Moderate	216	216	216
	Apple Scab High	197	197	197
Apple Rust Leaf	Apple Rust Low	216	216	216
	Apple Rust Moderate	203	203	203
	Apple Rust High	211	211	211
Total Images		1260	1260	1260

Table 6 Training, testing, and validation data for multistage leaf disease classification using plantvillage dataset without background images

Type of disease class	Stage of the disease	Train	Test	Validation
Apple Scab Leaf	Apple Scab Low	221	221	221
	Apple Scab Moderate	199	199	199
	Apple Scab High	202	202	202
Apple Rust Leaf	Apple Rust Low	187	187	187
	Apple Rust Moderate	227	227	227
	Apple Rust High	216	216	216
Total Images		1252	1252	1252

The primary goal of this study is to determine the model's optimal state for the 55 detection of plant leaf diseases and to perform multistage classification based on 8 disease severity levels. The experimental outcomes include training on both the original datasets and augmented datasets, making the model well-suited for real-time predictions.

4.1 Performance Evaluation: "The evaluation of the trained model's performance is conducted through the utilization of a confusion matrix, as depicted in Figure 7. This assessment involves the comparison of actual values with predicted values, leading to their categorization into distinct classes.""In the context of training for leaf disease detection using YOLOv5, we calculated the mean Average Precision (mAP) values for various classes:"

- Apple scab: mAP of 0.551
- Apple leaf (leaflet): mAP of 0.258

• Apple rust: mAP of 0.245

These values were obtained at a confidence score threshold of 0.7.

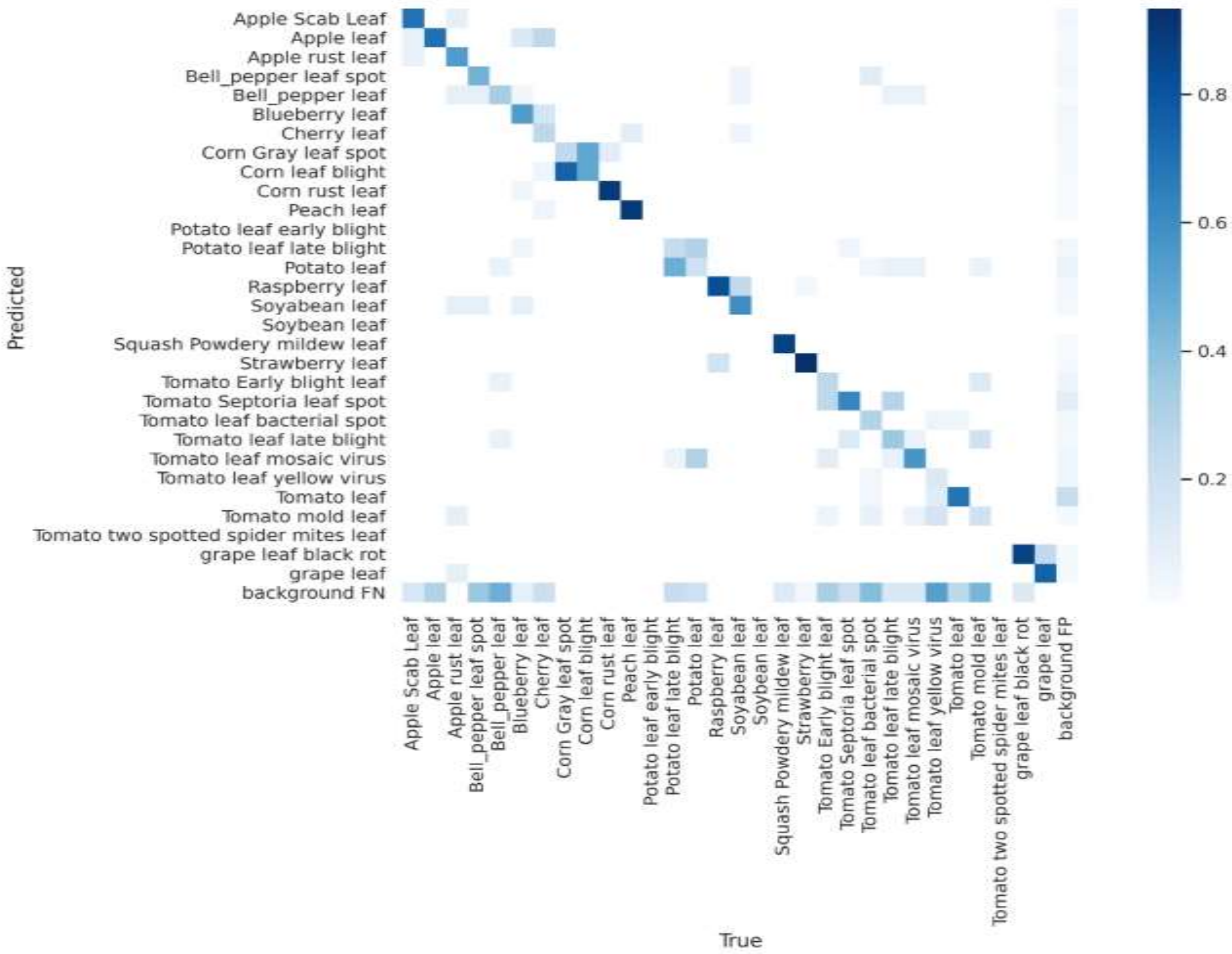
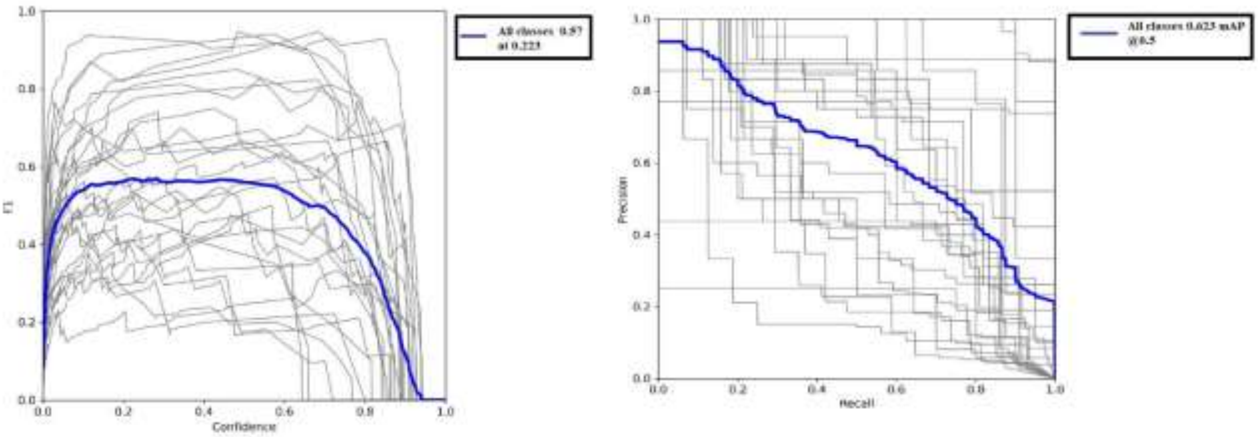


Figure 7 "Confusion matrix for leaf disease detection using the PlantDoc dataset."

(a) Leaf Disease Detection Training Performance: "The performance assessment for leaf disease detection involves the utilization of the F1 score as a statistical metric, which is applicable across different points on the Receiver Operating Characteristic (ROC) curve. Achieving a high F1 score necessitates both high precision and recall values.Figure 8(a) displays the F1 curve for training in leaf disease detection, encompassing 27 distinct classes from the PlantDoc dataset. On the x-axis, you can observe the confidence score, while the y-axis represents the F1 score. In the graph, various shades of grey depict the F1 scores for individual classes, while the blue line represents the overall F1 score for all classes. The highest observed F1 score is 0.57, achieved at a confidence score of 0.223.Moreover, precision and recall curves, commonly referred to as PrecisionRecall (PR) curves, are employed to assess performance at different probability thresholds. PR curves take into account class imbalance and play a critical role in evaluating model performance. The PR curve demonstrates a decreasing trend, signifying that lower thresholds lead to an increase in false positive predictions, while higher thresholds result in more false negatives. The PR-score is recorded at 0.623, accompanied by a mean Average Precision (mAP) of 0.5, as depicted in Figure 8(b)."

(b) Accuracy and Loss Graphs: "To gain a more comprehensive understanding of the performance in leaf disease detection using YOLOv5, various loss components are analyzed. These components encompass classification loss (cls_loss), objectness loss (obj_loss), and bounding box regression loss (box_loss), with their respective values detailed in Table 7."Given the presence of multiple classes in the dataset, the classification error is calculated to be 0.006629. Precision, which measures the percentage of accurate bounding box forecasts, and recall, which assesses how much of the true bounding box was accurately predicted, are determined to be 0.66002 and 0.54152, respectively. Mean Average Precision (mAP) is a critical evaluation metric in object detection models. At a confidence threshold of 0.5 for Intersection over Union (IoU), mAP 0.5 is calculated to be 0.61814. Furthermore, the average mAP over various IoU thresholds, ranging from 0.5 to 0.95, is represented as mAP 0.5:0.95, with a value of 0.46136. "The validation or testing metrics encompass box_loss (error), obj_loss (error), and cls_loss (error), with corresponding values of 0.031752, 0.011021, and 0.041854, respectively. These values are also visually depicted in multiple graphs, including training loss, accuracy metrics, and testing or validation loss, as presented in Figure 9."



a) F1 curve for PlantDoc Dataset PlantDoc b) PR curve for

Figure 8 :F1 curve and PR curve of leaf disease detection training

Table 7 Accuracy and loss values of leaf disease detection

Accuracy/Loss	**Value**
Training/box_loss(error)	0.028874
Training/obj_loss(error)	0.03059
Training/cls_loss(error)	0.006629
Metrics/precision	0.66002
Metrics/recall	0.54152
Metrics/mAP_0.5	0.61814
Metrics/mAP_0.5:0.95	0.46136
Validation/box_loss(error)	0.031752
Validation/obj_loss(error)	0.011021
Validation/cls_loss(error)	0.041854

Figure 9: Accuracy and loss graph of leaf disease detection training

(**c) Leaf Disease Detection Testing Analysis**: The testing analysis for leaf disease detection is visualized using a boxplot, providing a representation of confidence scores at thresholds of 0.3, 0.5, and 0.7 on the x-axis and Mean Average Precision (mAP) on the y-axis, as illustrated in Figure 10. The confidence score, also Known as the classification threshold, signifies the degree of confidence the machine learning model has in assigning the correct intent. It typically ranges from 0 to 1, with each user input assigned a score for each intent, and the highest score determining the outcome. The analysis reveals that the best Average mAP is achieved at 0.443, corresponding to a confidence score of 0.7. This result is notable as there are no outliers, and the Interquartile Range (IQR) is high. Table 8 presents the confidence score values and their corresponding average mAP at different thresholds. The box confidence score represents the minimum confidence threshold at which the model successfully detects an object.

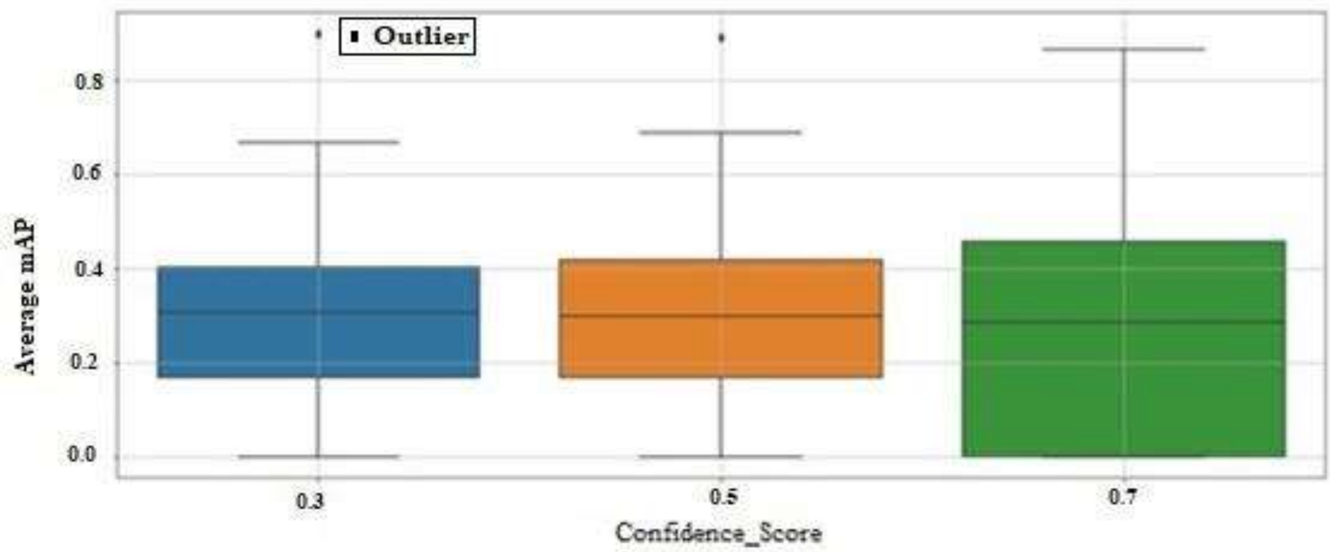


Figure 10 :Box plot of leaf disease detection testing on plantdoc dataset

Table 8 Inter quartile range of average map for different confidence scores of leaf disease detection testing

S. No.	Confidence score	Average mAP
1	0.3	0.2345
2	0.5	0.2593
3	0.7	0.443

4.2 Performance Analysis of Background Removal Technique:

To assess the effectiveness of the background removal technique, a manual evaluation is conducted on two classes of apple diseased leaves, namely apple scab and apple rust, utilizing the U2-Net architecture. The objective is to determine the quality of background removal. The process involves annotating the input image

using an open-source labeling tool named LabelMe, resulting in the generation of a binary image, as depicted in Figure 11.
"The Dice score, alternatively known as the Dice coefficient, serves as a metric used to quantify the effectiveness of background removal. This statistical measure evaluates the similarity between two datasets. In the context of this research, the Dice score is calculated for the two categories of apple diseased leaves by summing the intersection of both images and dividing it by the sum of the two images."

The obtained Dice scores are as follows:

- For apple scab: 0.750 +- 0.097
- For apple rust: 0.756 +- 0.093

These scores are represented in a box plot, as depicted in Figure 12.[Note: Proper attribution and citation should be provided when referring to specific evaluation metrics, techniques, or external tools used in your research, such as the LabelMe tool.

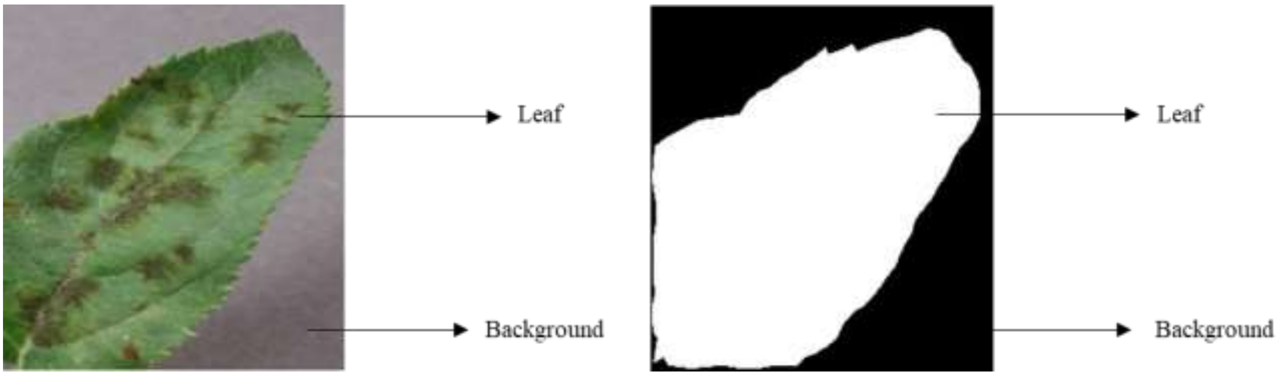


Figure 11 :Apple scab leaf RGB image and binary image with background

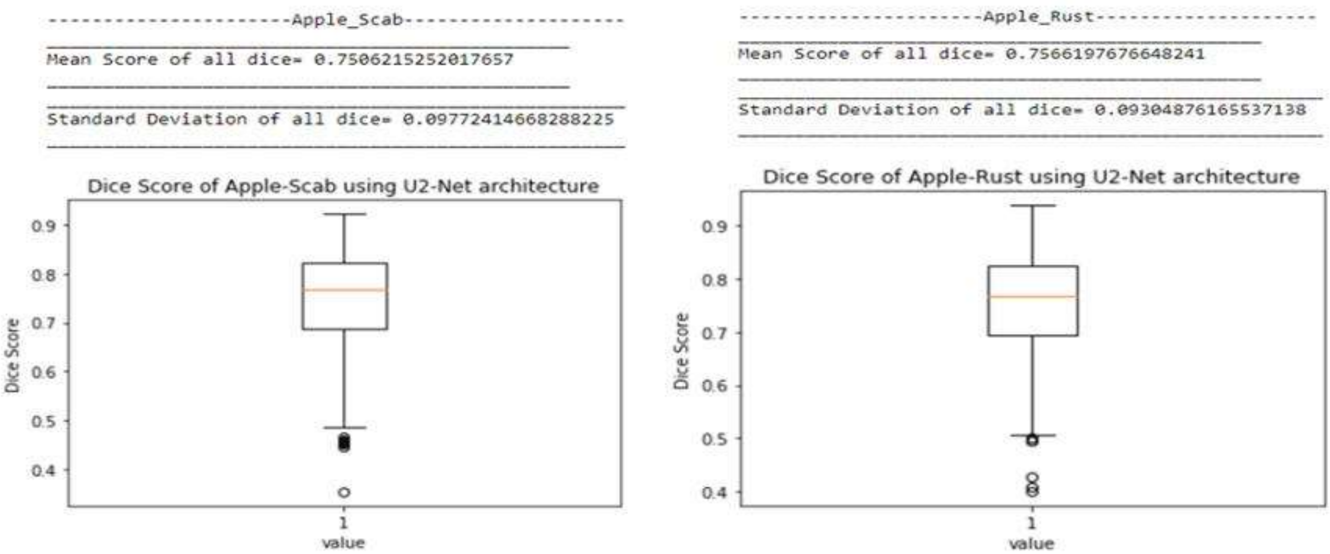


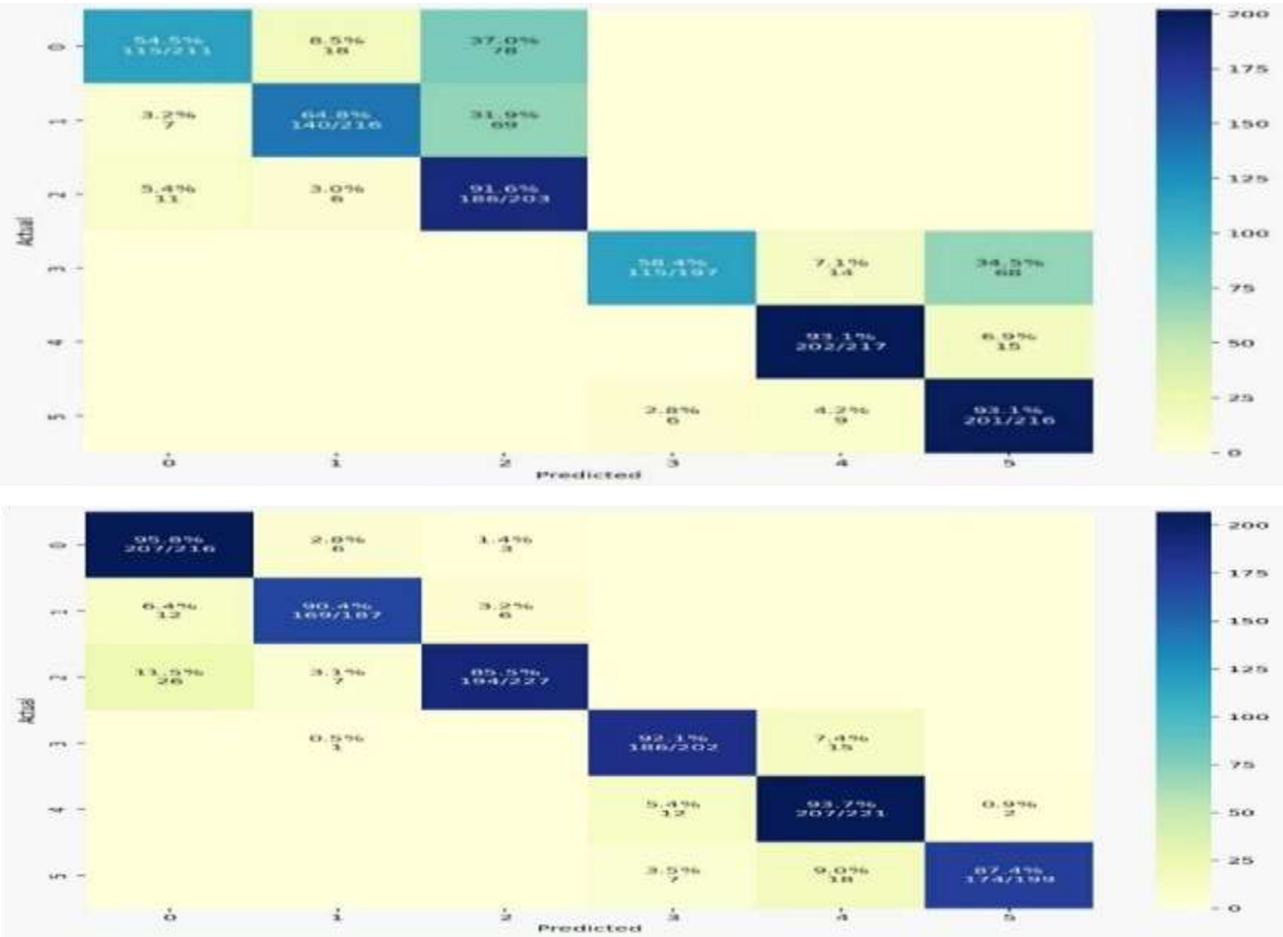
Figure 12: Dice score of apple scab and apple rust using U2-Net

4.3 Performance Analysis of Multistage Leaf Disease Classification: The performance of the multistage leaf disease classification model is assessed using various classification metrics to gauge its effectiveness. These metrics include:
1. Accuracy: It is computed as the proportion of correct predictions to The overall count of predictions.
2. Confusion Matrix: This table provides a comparison of model predictions with the ground truth.
3. Precision: Precision measures the ratio of correct positive predictions to all positive predictions made by the classifier.

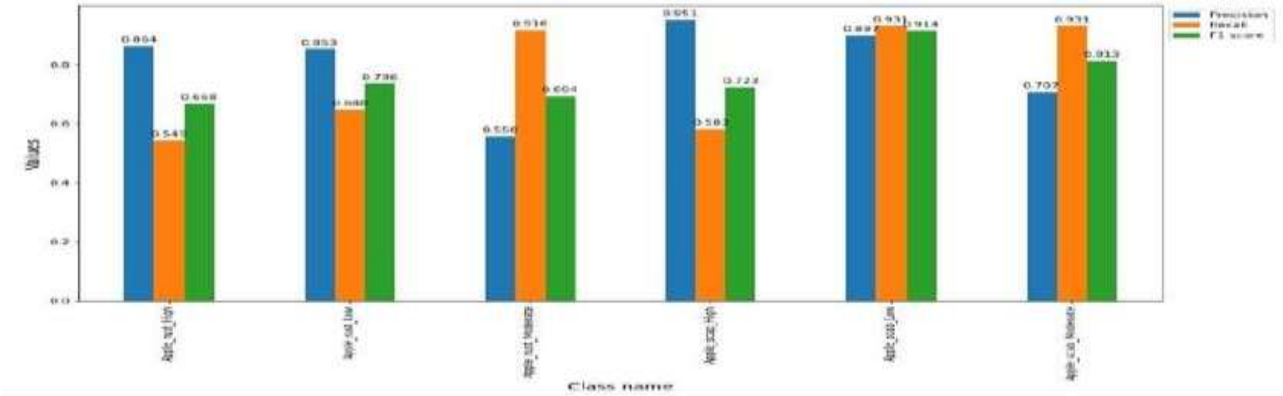
4. Recall: Recall is a metric that quantifies the proportion of actual positive cases in the dataset that the classifier correctly predicted. 5. F1 Score: The F1 score is the weighted average of Precision and Recall, considering both false positives and false negatives, as represented Within Equation 5.

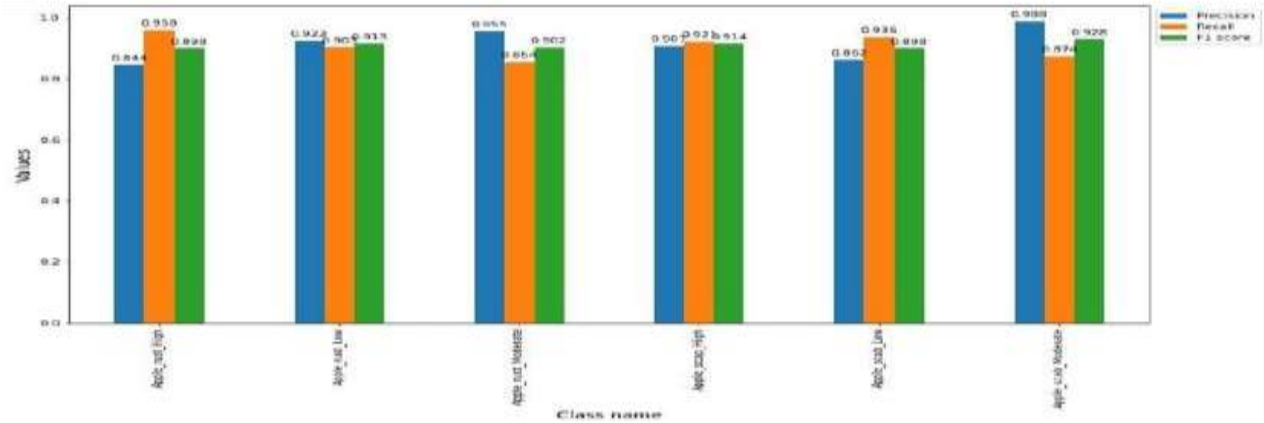
$$\text{F1 Score} = 2 \times \frac{Recall * Precision}{Recall + Precision} \quad (5)$$

These measures are employed to present the performance results of the multistage leaf disease classification model. The confusion matrix illustrates the model's output and is summarized in Figure 13. The overall F1 scores, with and without background removal, are reported as 0.75842 and 0.90846, respectively, as depicted in Figure 14. "4.4 Integration of Leaf Disease Detection and Severity Classification: In the integration model, an input image undergoes a series of processing steps to detect and identify the specific disease type, such as apple rust or apple scab. Subsequently, the contour of the image is highlighted, providing a visual representation of the object's boundaries. Contour images hold significance for shape analysis, object detection, and recognition. Following this, a heatmap image is generated, utilizing color coding to represent data intensity or density within a two-dimensional matrix or table. In a heatmap, each cell in the matrix is assigned a color based on its value, often following a gradient from low to high intensity. Leaf severity is determined by evaluating the percentage of infection on individual leaves using the Vision Transformer (ViT) classifier. This model has been tested for two categories of apple leaf diseases: apple rust and apple scab. The comprehensive pipeline for leaf disease detection and classification, encompassing severity assessment and recommendation solutions, is illustrated in Figures 15 and 16."



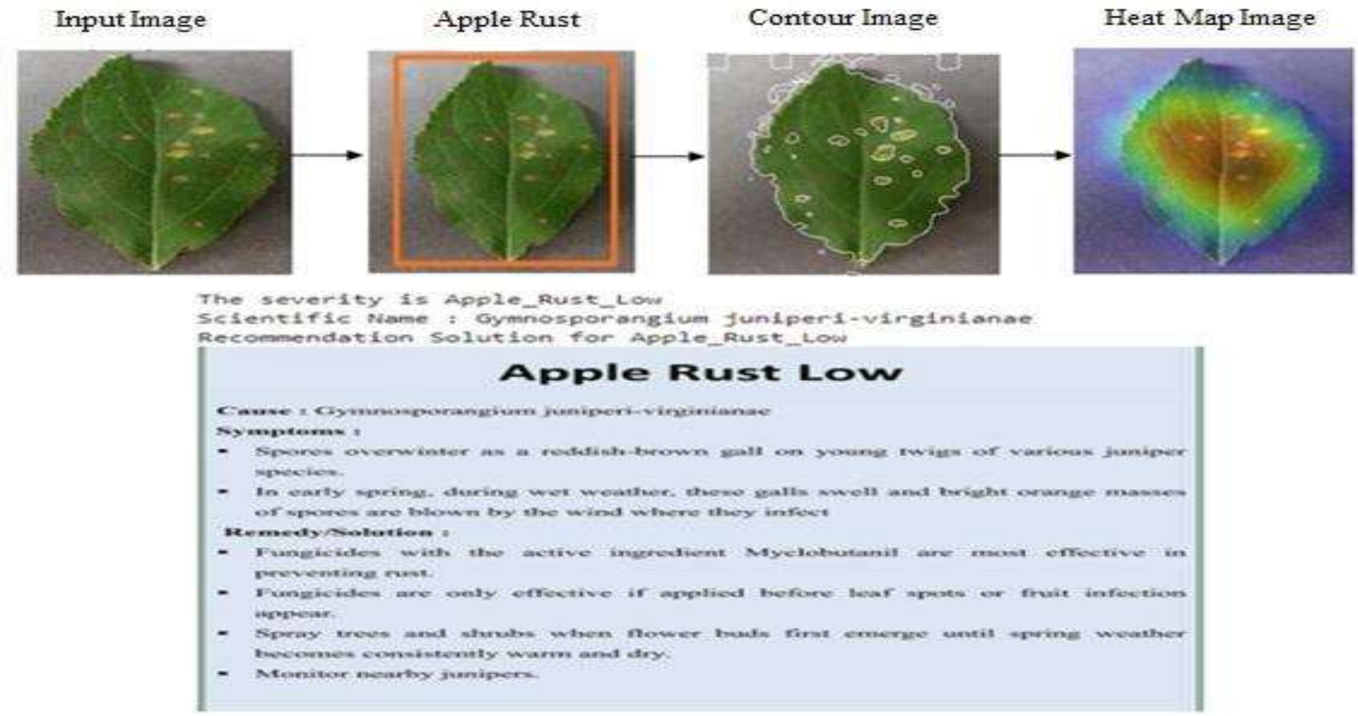
"Figure 13 displays the confusion matrix for multi-stage leaf disease classification, comparing results with and without background inclusion."





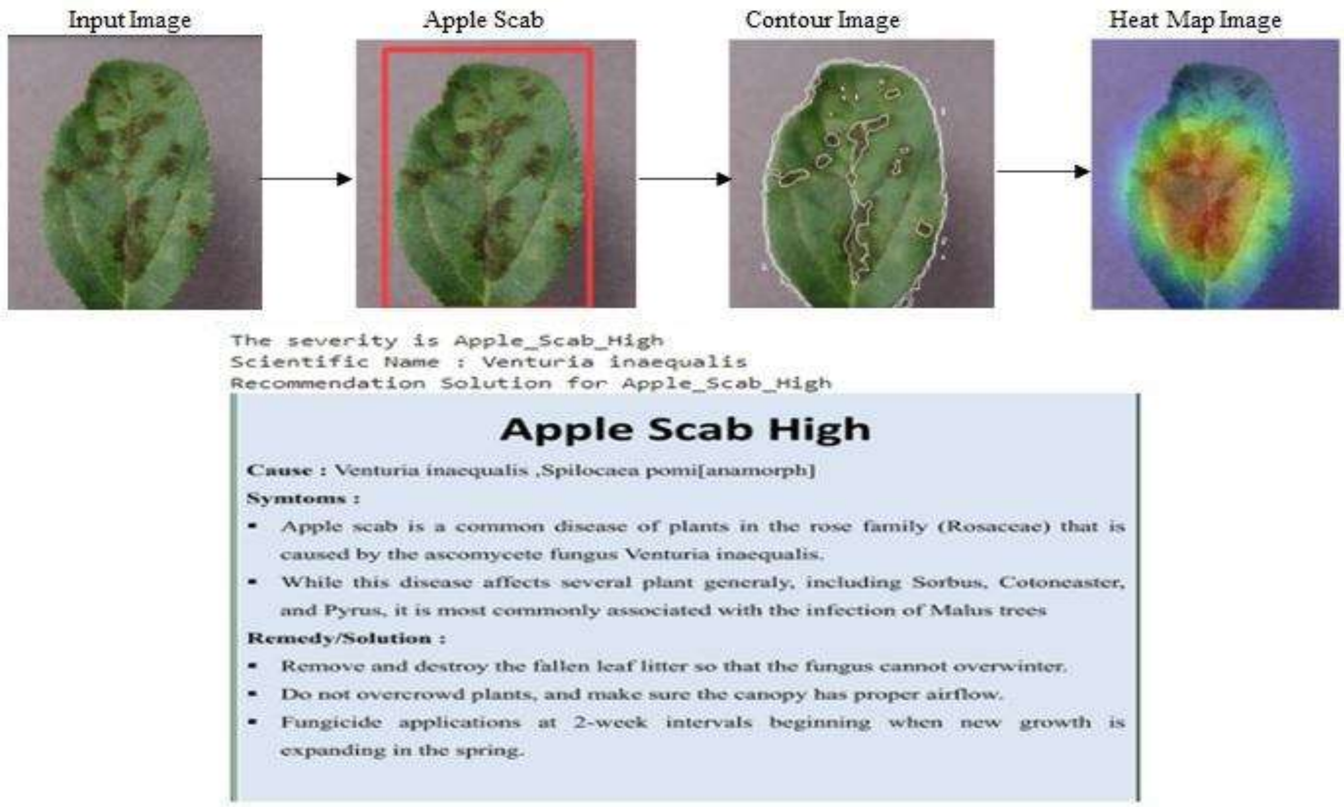
"Figure 14 illustrates the precision, recall, and F1 score for multi-stage leaf disease classification, providing a comparison between results with and without background inclusion."

4.5 Comparative Analysis: In this section, we provide a comparative evaluation of the performance concerning two distinct categories of apple leaf diseases, namely apple rust and apple scab, across various severity levels, including low, moderate, and high."We have calculated evaluation metrics such as precision, recall, and F1 score for each severity level, with due consideration for the support value. Table 9 and Table 10 present the performance values for multistage leaf disease classification with and without background removal. The overall F1 score for classification with background is reported as 0.758, while without



background, it is 0.910. "Table 11 presents a comparison of F1 scores for multi-stage leaf disease classification using the Vision Transformer (ViT) model, considering both scenarios with and without background inclusion."

Figure 15 :End-to-end leaf disease detection and classification results for apple rust



"Figure 16 illustrates the results of end-to-end leaf disease detection and classification specifically for apple scab"

Table 9 Precision, recall, and F1 score of Multistage Leaf Disease Classification with background

S. No.	Classes	Precision	Recall	F1 score	Support
1	Apple_rust_High	0.854	0.545	0.668	115
2	Apple_rust_Low	0.853	0.648	0.736	140
3	Apple_rust_Moderate	0.556	0.916	0.694	186
4	Apple_Scab_High	0.951	0.583	0.723	115
5	Apple_Scab_Low	0.897	0.931	0.919	202
6	Apple_Scab_Moderate	0.707	0.931	0.813	201
	Accuracy			0.758	959

Table 10 Precision, recall, and F1 score of multistage leaf disease classification without background

S. No.	Classes	Precision	Recall	F1 score	Support
1	Apple_rust_High	0.844	0.958	0.898	207
2	Apple_rust_Low	0.923	0.903	0.913	169
3	Apple_rust_Moderate	0.955	0.903	0.913	194
4	Apple_Scab_High	0.902	0.921	0.914	186
5	Apple_Scab_Low	0.862	0.936	0.898	207
6	Apple_Scab_Moderate	0.988	0.874	0.928	174
	Accuracy			0.910	1137

Table 11 Comparison of F1 Score for multistage leaf disease classification with and without background using Vi

S. No.	Model	Dataset used	F1 score
1	Multistage classification using ViT with background	Apple leaf disease from plantvillage dataset	0.758
2	Multistage classification using ViT without background	Apple leaf disease from plantvillage dataset	0.910

5. Discussion: In this section, we will delve into an in-depth discussion centered on the findings 28 derived from the preceding sections. The experimental investigation conducted within this study encompasses diverse facets:

1. Disease Detection: For disease detection, we utilized the YOLOv5 deep learning model. The outcomes demonstrated that, at a confidence score of 0.7, the model attained an average mAP (mean Average Precision) of 0.551 for apple scab, 0.258 for apple leaf, and 0.245 for apple rust classifications. In addition to mAP, we employed the F1 score and PR curve as performance evaluation metrics.

2. Background Removal: The U2-Net architecture played a crucial role in enhancing disease classification accuracy by effectively removing the background. The assessment of background removal performance involved the 6 use of the Dice score, a statistical measure, which yielded Dice scores of 0.750±0.097 for apple scab and 0.756±0.093 for apple rust.

3. Leaf Disease Stage Classification: The classification of leaf disease severity was conducted through the utilization of the Vision Transformer (ViT) classifier. Performance evaluation included metrics such as accuracy, recall, precision, F1 score, and other relevant indicators.

4. Comparative Analysis: A comparative analysis was conducted, considering precision, recall, and F1 score, with and without background removal. The results showed that classification without background outperformed with an overall F1 score of 0.910.

Implications: This research holds several implications:

• "Enhanced Accuracy: The approach proposed, grounded in deep learning, showcases a remarkable level of accuracy in the classification of plant leaf disease severity. This achievement can lead to more effective disease management, potentially bolstering crop yields and mitigating economic losses."

• Explainability in AI: Utilization of explainable AI (XAI) techniques, such as PR curves and Grad-CAM, enhances transparency and trust in AI models, providing insights into their decision-making processes.

• "Applicability of Transformer Models: The effectiveness of transformerbased models in leaf disease classification implies their potential use in various other image classification tasks.

• Advancements in Plant Disease Detection: The adoption of deep learning in plant disease detection offers the potential to revolutionize agriculture by enabling early disease identification and efficient management."

Limitations: "This research is concentrated exclusively on the severity classification of two specific apple leaf diseases, which are apple rust and apple scab. It does not encompass a broader range of plant species or diseases."

6. Conclusion and Future Work: "The primary aim of this study was to create a sophisticated deep learning model designed for the detection of plant leaf diseases and the classification of their severity levels, emphasizing originality. The research yielded several key findings."

• Using disease detection software YOLOv5 achieved high accuracy, with a maximum detection accuracy of 0.443 mAP at a confidence score of 0.7.

• Background removal using the U2-Net architecture improved disease classification accuracy, as evidenced by Dice scores.

• Severity classification using the ViT classifier was effective, and an end-to-end application was developed for real-time use.

• The proposed system provides recommendations for disease mitigation based on classification and analysis.

Future Work: Future work can address the following areas:

• Improved Accuracy: Further improvements in accuracy on real-time datasets can be pursued.

• Expansion to Other Plant Species: The system can be extended to include different plant species and diseases, which may lead to significant accuracy enhancements.

• Data Augmentation: Additional training data and data augmentation techniques can be explored to further improve model performance. "In conclusion, this study highlights the promise of deep learning, explainable artificial intelligence (XAI), and transformer-based models in the realm of plant disease detection and management. These findings have broader implications for agricultural research and beyond."

References:

[1] Ministry of Agriculture and Farmers' Welfare. (2020-21). Annual Report. Retrieved from https://agricoop.nic.in/Documents/annual-report-2020-21.pdf. Accessed on 19 February 2023.

[2] Raza, A., Razzaq, A., Mehmood, S. S., Zou, X., Zhang, X., Lv, Y., et al. (2019). Impact of Climate Change on Crops: Adaptation and Strategies to Tackle Its Outcome – A Review. Plants, 8(2), 1-29.
[3] López, M. M., Bertolini, E., Olmos, A., Caruso, P., Gorris, M. T., Llop, P., et al. (2003). Innovative Tools for Detection of Plant Pathogenic Viruses and Bacteria. International Microbiology, 6, 233-243.
[4] Liang, Q., Xiang, S., Hu, Y., Coppola, G., Zhang, D., Sun, W. (2019). PD2SENet: Computer-Assisted Plant Disease Diagnosis and Severity Estimation Network. Computers and Electronics in Agriculture, 157, 518-529.
[5] Singh, D., Jain, N., Jain, P., Kayal, P., Kumawat, S., Batra, N. (2020). PlantDoc: A Dataset for Visual Plant Disease Detection. In Proceedings of the 7th ACM IKDD CoDS and 25th COMAD 2020 (pp. 249-253).
[6] Kaggle. (n.d.). Plant Village Dataset. Retrieved from https://www.kaggle.com/datasets/abdallahalidev/plantvillage-dataset. Accessed on 10 February 2023.
[7] Fang, Y., Ramasamy, R. P. (2015). Current and Prospective Methods for Plant Disease Detection. Biosensors, 5(3), 537-561. [8] Mou, L., Meng, Z., Yan, R., Li, G., Xu, Y., Zhang, L., et al. (2016). How Transferable Are Neural Networks in NLP Applications? In Proceedings of the Conference on Empirical Methods in Natural Language Processing 2016 (pp. 479- 489). Association for Computational Linguistics.
[9] Dhanya, V. G., Subeesh, A., Kushwaha, N. L., Vishwakarma, D. K., Kumar, T. N., Ritika, G., et al. (2022). Deep Learning-Based Computer Vision Approaches for Smart Agricultural Applications. Artificial Intelligence in Agriculture, 6, 211-219.
[10] Dhaygude, S. B., Kumbhar, N. P. (2013). Agricultural Plant Leaf Disease Detection Using Image Processing. International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 2(1), 599-602.
[11] Sethy, P. K., Barpanda, N. K., Rath, A. K. (2019). Detection and Identification of Rice Leaf Diseases Using Multiclass SVM and Particle Swarm Optimization Technique. International Journal of Innovative Technology and Exploring Engineering, 8(6S2), 2278-3075.
[12] Sebastian, V. B., Unnikrishnan, A., Balakrishnan, K. (2012). Gray Level CoOccurrence Matrices: Generalization and Some New Features. International Journal of Computer Science, Engineering and Information Technology, 2(2), 151-157.
[13] Khirade, S. D., Patil, A. B. (2015). Plant Disease Detection Using Image Processing. In International Conference on Computing Communication Control and Automation 2015 (pp. 768-771). IEEE.
[14] Cao, D., Chen, Z., Gao, L. (2020). An Improved Object Detection Algorithm Based on Multi-Scaled and Deformable Convolutional Neural Networks. Humancentric Computing and Information Sciences, 10(1), 1-22.
[15] Guo, Y., Zhang, J., Yin, C., Hu, X., Zou, Y., Xue, Z., et al. (2020). Plant Disease Identification Based on Deep Learning Algorithm in Smart Farming. Discrete Dynamics in Nature and Society, 2020, 1-11.
[16] Cohen, R. (2011). The Chan-Vese Algorithm. Computer Vision and Pattern Recognition, 1-18.
[17] Hassan, S. M., Maji, A. K., Jasiński, M., Leonowicz, Z., Jasińska, E. (2021). Identification of Plant-Leaf Diseases Using CNN and Transfer-Learning Approach. Electronics, 10(12), 1-19.
[18] Rashid, J., Khan, I., Ali, G., Almotiri, S. H., Alghamdi, M. A., Masood, K. (2021). Multi-Level Deep Learning Model for Potato Leaf Disease Recognition. Electronics, 10(17), 1-27.
[19] Redmon, J., Divvala, S., Girshick, R., Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the Conference on Computer Vision and Pattern Recognition 2016 (pp. 779-788). IEEE.
[20] Redmon, J., Farhadi, A. (2017). YOLO9000: Better, Faster, Stronger. In Proceedings of the Conference on Computer Vision and Pattern Recognition 2017 (pp. 7263-7171). IEEE.
[21] Redmon, J., Farhadi, A. (2018). YOLOv3: An Incremental Improvement. Computer Vision and Pattern Recognition, 1-6.
[22] Bochkovskiy, A., Wang, C. Y., Liao, H. Y. (2020). YOLOv4: Optimal Speed and Accuracy of Object Detection. Computer Vision and Pattern Recognition, 1-17.
[23] Ultralytics. (n.d.). YOLOv5 GitHub Repository. Retrieved from https://github.com/ultralytics/yolov5. Accessed on 10 February 2023.

[24] Ren, S., He, K., Girshick, R., Sun, J. (2016). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1-14.
[25] Pascal VOC. (n.d.). Pascal VOC 2007. Retrieved from http://www.pascalnetwork.org/challenges/VOC/voc2007/index.html. Accessed on 15 February 2023.
[26] Pascal VOC. (n.d.). Pascal VOC 2012. Retrieved from http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2012/index.html. Accessed on 15 February 2023.
[27] Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., et al. (2014). Microsoft COCO: Common Objects in Context. In Computer Vision–ECCV, 13th European Conference, Zurich, Switzerland, Proceedings 2014 (pp. 740-755). Springer International Publishing.
[28] Tan, M., Le, Q. (2019). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In International Conference on Machine Learning 2019 (pp. 6105-6114). PMLR.
[29] Malik, H. S., Dwivedi, M., Omkar, S. N., Javed, T., Bakey, A., Pala, M. R., et al. (2021). Disease Recognition in Sugarcane Crop Using Deep Learning. In Advances in Artificial Intelligence and Data Engineering 2021 (pp. 189-206). Springer Singapore. [30] Toda, Y., Okura, F. (2019). How Convolutional Neural Networks Diagnose Plant Disease. Plant Phenomics, 2019, 1-14.
[31] Nagasubramanian, K., Jones, S., Singh, A. K., Sarkar, S., Singh, A., Ganapathysubramanian, B. (2019). Plant Disease Identification Using Explainable 3D Deep Learning on Hyperspectral Images. Plant Methods, 15, 1-10.
[32] Liu, J., Wang, X. (2020). Early Recognition of Tomato Gray Leaf Spot Disease Based on MobileNetV2-YOLOv3 Model. Plant Methods, 16, 1-6.
[33] Malik, A., Vaidya, G., Jagota, V., Eswaran, S., Sirohi, A., Batra, I., et al. (2022). Design and Evaluation of a Hybrid Technique for Detecting Sunflower Leaf Disease Using Deep Learning Approach. Journal of Food Quality, 2022, 1-12.
[34] Tammina, S. (2019). Transfer Learning Using VGG-16 with Deep Convolutional Neural Network for Classifying Images. International Journal of Scientific and Research Publications, 9(10), 143-150.
[35] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L. C. (2018). MobileNetV2: Inverted Residuals and Linear Bottlenecks. In Proceedings of the Conference on Computer Vision and Pattern Recognition 2018 (pp. 4510-4520). IEEE.
[36] Srinidhi, V. V., Sahay, A., Deeba, K. (2021). Plant Pathology Disease Detection in Apple Leaves Using Deep Convolutional Neural Networks: Apple Leaves Disease Detection Using EfficientNet and DenseNet. In 5th International Conference on Computing Methodologies and Communication 2021 (pp. 1119-1127). IEEE.
[37] Sibiya, M., Sumbwanyambe, M. (2021). Automatic Fuzzy Logic-Based Maize Common Rust Disease Severity Predictions with Thresholding and Deep Learning. Pathogens, 10(2), 1-17.
[38] Shiferaw, D., Hailegnaw, A., Assefa, A., Abebe, D., Dagne, E., Fekadie, G. (2019). LabVIEW-Based Fuzzy Logic Controller for PLC. In International Conference on Computer, Control, Electrical, and Electronics Engineering 2019 (pp. 1-4). IEEE.
[39] Shi, T., Liu, Y., Zheng, X., Hu, K., Huang, H., Liu, H., Huang, H. (2023). Recent Advances in Plant Disease Severity Assessment Using Convolutional Neural Networks. Scientific Reports, 13(1), 1-14.
[40] Abd, A. Y. M., Caro, O. J., Bravo, L. M., Kaur, C., Al AMS, Bala, B. K. (2023). Leaf Disease Identification and Classification Using Optimized Deep Learning. Measurement: Sensors, 25, 1-6.
[41] Mohammed, L., Yusoff, Y. (2023). Detection and Classification of Plant Leaf Diseases Using Digital Image Processing Methods: A Review. ASEAN Engineering Journal, 13(1), 1-9.
[42] Han, K., Wang, Y., Chen, H., Chen, X., Guo, J., Liu, Z., et al. (2022). A Survey on Vision Transformer. IEEE Transactions on Pattern Analysis and Machine Intelligence, 45(1), 87-110.
[43] Xu, R., Lin, H., Lu, K., Cao, L., Liu, Y. (2021). A Forest Fire Detection System Based on Ensemble Learning. Forests, 12(2), 1-16.
[44] Bo, S., Jing, Y. (2012). Image Clustering Using Mean Shift Algorithm. In Proceedings of the Fourth International Conference on Computational Intelligence and Communication Networks 2012 (pp. 327-330). IEEE.

[45] Canny, J. (1986). A Computational Approach to Edge Detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, 8(6), 679-698.
[46] Qin, X., Zhang, Z., Huang, C., Dehghan, M., Zaiane, O. R., & Jagersand, M. (2020). U2-Net: going deeper with nested U-structure for salient object detection. Pattern Recognition, 106, 1-12.
[47] Arvind, C. S., Prajwal, R., Bhat, P. N., Sreedevi, A., & Prabhudeva, K. N. (2019). Fish detection and tracking in pisciculture environment using deep instance segmentation. In Proceedings of the TENCON Region 10 Conference 2019 (pp. 778- 783). IEEE.

Navyashree KS currently serves as an Assistant Professor in the Department of Computer Science and Engineering, specializing in Data Science, at RNSIT in Bangalore, India. She attained her M.Tech in Computer Science and Engineering from Dr. Ambedkar Institute of Technology, Bangalore, in 2022, following her Bachelor of Engineering from SJB Institute of Technology, Bangalore, in 2019. She harbors a fervent enthusiasm for utilizing data to benefit society, with a particular focus on AI, Data Science, Machine Learning, and Computer Vision technologies. Email:snavya447@gmail.com.

Ravi Kiran K V presently holds the position of a Data Scientist. He completed his postgraduate studies, earning a Master of Technology in Computer Science & Engineering from Dr. Ambedkar Institute of Technology, Bangalore, in 2022, accredited by Visvesvaraya Technological University (VTU), Belgaum, Karnataka. Prior to this, he attained his Bachelor of Engineering in Information Science and Technology from East West Institute of Technology, Bangalore in 2020. His expertise plays a crucial role in driving technological advancements in Artificial Intelligence and Data Science. Email: kv.kiran7@gmail.com