AVE Trends in Intelligent Computer Letters

Designing a Sustainable Agriculture Platform for Farm Monitoring and Peach Detection Using Advanced Artificial Intelligence Techniques

S. Gopikha[1,*], V. S. Anusuya[2], Thirumalraj Karthikeyan[3], Vijilius Helena Raj[4], Sureshkumar Somayajula[5]

[1]Department of Information Technology, St. Joseph's College of Engineering, Chennai, Tamil Nadu, India.
[2]Department of Chemistry, New Horizon College of Engineering, Bengaluru, Karnataka, India.
[3]Department of Artificial Intelligence, Trichy Research Labs, Quest Technologies, Tiruchirappalli, Tamil Nadu, India.
[4]Department of Mathematics, New Horizon College of Engineering, Bengaluru, Karnataka, India.
[5]Department of Computer Science and Technology, Sunlife Canada Financials, Toronto, Ontario, Canada.
gopikha.re@gmail.com[1], anukmp@gmail.com[2], thirumalraj.k@gmail.com[3], vijilius@gmail.com[4], suresh.kumar.somayajula@sunlife.com[5]

*Corresponding author

Abstract: Considering the difficulties in identifying pear leaf diseases due to factors such as varying lighting conditions, overlapping leaves, and other green plants in the background, a two-stage framework based on strategies is suggested for segmenting pear leaves and disease categorisation. To begin, the target damaged pear leaf is extracted, and background interference is eliminated by building a DBPN that fuses the low-level feature branch and the semantic branch. Research on transformer-based models has grown in recent years, with several studies showing promising results. Unfortunately, transformers still have issues with edge detail segmentation and small object recognition. To address these issues, the research enhanced the Swin transformer by combining the best features of transformers and CNNs. It then developed an LPSW backbone to boost the network's local perception and classification task discovery accuracy. A framework for an SAIEC system was also developed as part of the study; this contributed to the network's enhanced accuracy. To fine-tune the parameters of the proposed classifier, this paper proposes using improved beetle swarm optimisation. The field dataset DiaMOS Plant, comprising 3505 images of pear fruit and leaves damaged by four illnesses, is used to evaluate the proposed model in this study. The dataset is publicly available and was collected to identify and monitor plant problems. both segmentation and classification.

Keywords: Spatial Attention Interleaved Execution Cascade (SAIEC); Local Perception Swin transformer (LPSW); Edge Detail Segmentation; Beetle Swarm Optimisation; Pear Leaf Disease Detection; Convolutional Neural Network (CNN); Double-Branch Polymerisation Net (DBPN).

Cite as: S. Gopikha, V. S. Anusuya, T. Karthikeyan, V. H. Raj, and S. Somayajula, "Designing a Sustainable Agriculture Platform for Farm Monitoring and Peach Detection Using Advanced Artificial Intelligence Techniques," *AVE Trends in Intelligent Computer Letters*, vol. 1, no. 3, pp. 119–131, 2025.

Journal Homepage: https://avepubs.com/user/journals/details/ATICL

Received on: 15/08/2024, **Revised on:** 30/09/2024, **Accepted on:** 21/11/2024, **Published on:** 05/09/2025

DOI: https://doi.org/10.64091/ATICL.2025.000227

1. Introduction

Sustainable agriculture methods may enhance product accessibility in a supply chain. Soil structure and an orchard's resilience to climate change, intense harvest production strains, and extreme weather events might be jeopardised by some conventional methods. When implementing sustainable agricultural techniques, it is essential to consider all approaches, whether new or

conventional [1]. One way to lessen the impact of global warming is to use precision agricultural methods. Because they can accurately predict products, deep neural networks are becoming powerful tools in precision agriculture. This leads to better management of resources, such as irrigation water and pesticides, and helps reduce food loss and waste by improving the scheduling of agricultural activities [2]. Precision agriculture includes yield estimation as one of its subfields. It enables the organisation of supplementary services (such as robot fruit pickup) alongside farming, operations, and inventory management. The topic of fruit yield estimation has been covered in many published papers. For the yield estimating method to work, the fruit identification step is crucial [3]. Aside from contributing to food production, agriculture also generates resources essential to producing goods for sale. Crops were cultivated using time-honoured agricultural methods. Nearly every country still uses some form of traditional farming. Methods proposed by seasoned farmers are a part of it. Because of their imprecision, these methods require significant manual labour [4]. Precision Agriculture refers to the use of digital technology, such as electronic devices, sensors, robotics, and automation. Workload reduction, increased profitability, and better decision-making are the goals of this technology [5].

To optimise profitability, maximise yield, and enhance production quality, precision agriculture—also known as precision farming—is an agricultural control system that provides a comprehensive approach to managing the spatial and temporal variability of crops and soil [6]. Increasing crop yields has never been easier than with precision agriculture. Precision agriculture was more often used by farms with high asset valuations than by farms with lower asset valuations, according to a discussion of the topic [7]. Country and geographical factors also have a role in the rate of precision agriculture adoption. Compared to farmers in the valley, those in the mountain zone are less likely to use Precision Agriculture. Substantial investments are required to increase adoption rates. For farms of all sizes to adopt precision agriculture, the cost of the necessary machinery must be reduced [8]. Data Science, Artificial Intelligence, Data Mining, and the Internet of Things all play a role in precision agriculture [9]. Agronomic applications are utilising wireless sensor networks to remotely monitor soil and ambient variables for crop health prediction. Agricultural fields can have their irrigation schedules anticipated using WSN as a forecasting method. Among the many environmental factors that wireless sensor networks monitor are soil moisture, salinity, temperature, and humidity [10]. The wonderful flavour and high nutritional content of pears make them a favourite fruit in China. Growth in the pear tree business is good for the local economy and the farmers who cultivate pears, who get substantial financial rewards as a result [11]. Natural environmental variables, however, can cause a variety of illnesses to manifest in the leaves of pear trees as they mature.

A significant drop in pear quality and a flood of infected pears caused devastating economic losses. Consequently, it is crucial to improve pear production and quality, and to increase farmers' economic returns by promptly identifying and preventing pear leaf diseases [12]. Manual plant disease diagnosis has a long history of use but is now fraught with issues such as high labour costs, low recognition accuracy, and high subjectivity. Advancements in computer technology and artificial intelligence facilitate a rise in agricultural output. To accomplish disease diagnosis tasks, intelligent agriculture develops feature extractors that use image processing to extract characteristics of diseased plant leaves [13]. Light, leaflet overlap, and stem occlusion often result in noisy pear leaf photos with complicated backgrounds. The characteristics of sick patches are difficult to extract due to these circumstances, resulting in low identification accuracy [14]. In many modern applications, the basis of state-of-the-art fruit detection systems is. Image attributes (such as colours and forms) of fruit may be automatically extracted using these approaches. Fruit detection has progressed in a context-specific manner due to the uniqueness of the fruits' sizes, shapes, colours, clusters, and tree distributions [15]. CNNs excel at extracting local information, but they can't handle global data with long-range dependencies. Many computer vision tasks suggest using self-attention techniques to efficiently overcome CNN limitations, drawing inspiration from the transformer's use of self-attention and the need to mine long-range correlations in text [16]. Self-attention systems can swiftly learn relationships between distant elements, attend to various parts of the image, and integrate data across the image. The main contributions of this paper can be abridged as follows:

- **Study on the Pear Leaf Segmentation Method:** To address the complex background and feature-scale issues in pear leaf images captured in the real production environment, a double-branch polymerisation network (DBPN) segmentation algorithm was proposed. DBPN used an asymmetric double–branch encoder–decoder framework to enhance multi-scale feature extraction. The low-level feature branch consisted of a three-branch spatial enhancement module (TBSEM) to establish pixel-level dependency relations, which enhanced the spatial focusing and detail reshaping capabilities of DBPN. The semantic branch used an atrous spatial pyramid pooling (ASPP) module to extract contextual information and achieve a sufficiently wide receptive field, thereby enabling the extraction of additional semantic features and improving segmentation accuracy.
- **Study on the Classification Task:** Researchers used the transformer as our foundational network to construct a network well-suited for classification, aiming to circumvent CNNs' limitations in extracting global information. To boost local perception skills and categorisation detection accuracy, the LPSW leverages the strengths of CNNs and transformers. Researchers present the SAIEC network architecture for spatial attention interleaved execution cascades. The network's mask prediction is improved by the multi-tasking method and the improved spatial attention

unit. Lastly, a new network model is established by inserting the LPSW into the developed framework as its backbone. This model significantly improves detection accuracy.

2. Related Works

Parez et al. [17] proposed an improved method for identifying plant illnesses and infections using Vision Transformers (ViTs), called GreenViT. Researchers give the ViT a series of smaller blocks, or patches, that represent the input image, much like word embeddings. Researchers circumvent the issues of CNN-based models by capitalising on ViTs' capabilities. To test how well the proposed GreenViT would work, researchers ran experiments on popular benchmark datasets. Results from experiments show that, compared to SOTA CNN models, the proposed method is superior at identifying plant illnesses. Rehman et al. [18] proposed a practical, workable method for producing reliable agricultural data using remote sensing. The 107,899 pixels that make up our self-collected dataset were split into 30% for testing and 70% for training. To prevent spatial autocorrelation, the acquired data is presented as field parcels, which are further split into training, validation, and test sets. The correctness and quality of the training data were checked by excluding 15% for validation and 15% for testing. Our trained model was also used for prediction, and the results were significant based on visual examination of the picture region. In addition, the combined Planet-Scope and Sentinel-2 time series data sets are not utilised to compare the series; the attained weighted average is 93%, and the merged Planet-Scope and Sentinel-2 time series 97%. For precision agriculture, Punithavathi et al. [19] propose a new model, CVDL-WDC, based on deep learning and computer vision. The goal of the proposed CVDL-WDC method is to accurately distinguish between plants and weeds. Both object identification using multiscale Faster R-CNN and weed classification using an optimal extreme learning machine (ELM) are part of the proposed CVDL-WDC method.

The farmland fertility optimisation (FFO) approach is used to fine-tune the ELM model's parameters. The improved results over its recent methods across many metrics were demonstrated by a thorough simulation study of the CVDL-WDC method using a benchmark dataset. A deep learning-based peach packing robot prototype was created by Wang et al. [20]. To begin, researchers constructed the peach object recognition dataset. Researchers used it to train end-to-end YOLO v5 models with varying width and depth, carefully balancing accuracy and real-time performance. Afterwards, the coordinate transformation matrix was computed using the "Eye-on-Base" hand-eye calibration method, which aligned the camera coordinate system with the robot base coordinate system. According to the findings of the landmark positioning experiment, the average positioning errors along the X- and Y-axes were 4.87 mm and 5.00 mm, respectively.

Mostly due to the RGB-D camera's depth estimation error, the average Z-direction placement error in the Z direction was 18.47 mm. With a success rate of 100% for little peaches, 97% for medium peaches, and 97% for giant peaches, the grasping experiment investigated the impact of accuracy. Depth perception, object identification, coordinate transformation, and grasping route planning took an average of 252.81 ms over the complete pipeline suggested in this study. Next, SFDI technology was used to assess early-stage bruising in peaches. Overall, this study laid out a solid plan for the fruit-packing robot, which could be used in post-harvest marketing. A method for determining peach maturity based on visible and tactile traits has been proposed by Wang et al. [21]. Two steps make up this method. In the initial step, YOLOv4 is used as the main model for initial visual categorisation of fruit ripeness.

For a more in-depth tactile ripeness classification, the second step suggests a system based on flexible piezoelectric sensors. The peach sorting line can use the system's categorisation of peaches into five groups: A1, A2, B1, B2, and R. The visual portion is evaluated using the mAP and meanIoU, whereas the tactile part is validated using the TPA test. Experimental findings reveal a mean IoU of 0.9454, a mAP of 0.9304, and tactile part accuracy of 92.22%. The findings show that optical and tactile traits, in conjunction with YOLOv4 and a flexible piezoelectric sensor, may reliably categorise peach ripeness. With the suggested method, researchers can automate the sorting line and complete classification at reduced cost and with lower power consumption. Teng et al. [22] constructed 3D models of peach tree trimming in winter using UAV-SfM and 3D lidar SLAM methods. Researchers next proposed a method to differentiate branches from 3D point clouds based on spatial density, after comparing and analysing these models. Compared with UAV-SfM, the 3D lidar SLAM method for winter peach tree trimming required less time to model and produced more accurate results. Compared with the initial weight of the trimmed branches, the method's RMSE was lowest at 3084 g, and its R2 was 0.93. Whether performed before or after pruning, the branch identification portion correctly identified branches with diameters greater than 3 cm. There are still major issues in the CV field, despite significant improvements in object recognition with transformer-based techniques:

- Weak local information collecting capabilities and poor identification performance for small-scale items.
- The current transformer-based architecture is primarily used for image categorisation; however, achieving satisfactory results for instance segmentation in densely annotated images is challenging for a single-level transformer. This significantly affects the accuracy of object recognition and request segmentation in high-resolution, complex-background, small-object remote sensing pictures.

3. Materials

Researchers provide a detailed explanation of the projected dataset here. Description. Here, researchers provide DiaMOS Plant, an expanded dataset examined in Fenu and Malloci [23], as a field dataset for plant symptom diagnosis and monitoring. A pilot dataset, DiaMOS Plant, was created to establish a representative sample that captures the key cultural features of pear trees. The dataset contains photos of a pear tree during its full growing season, from February to July. Machine learning and deep learning techniques may be applied to detection and classification problems with this dataset. The total sum of photographs collected is 3505. Out of them, 499 depict fruits and 3006 depict leaves. Here are four stages of fruit expansion: setting, nut fruit, growth, and ripening. Similarly, there are four types of biotic and abiotic stresses: healthy leaf, slug damage, curling leaves, and spotty leaves. You may find a comprehensive overview in Tables 1 and 2.

Table 1: Dataset explanation

DiaMOS Plant Dataset	Dataset Description
Plant	Pear
Annotation	csv, YOLO
Cultivar	Septoria Piricola
Data Source Site	Sardegna, Italy
Category of data	RGB Imageries
Total size	3505 imageries
Data Accessibility	Fenu and Malloci [32]

The DiaMOS plant image library contains 3,355 distinct fruit and leaf images. Classes associated with the leaf pictures are distributed as shown in Table 2.

Table 2: Leaf damage indicators by size

Leaf Indications	Size
Fit	43
Spot	848
Curl	54
Slug	2035

All three trees in the photos are from the same Italian land. A smartphone (Honour 6Ï) and a DSLR camera (Canon EOS 60D) were used to capture the photographs; as a result, the images have two distinct resolutions: 2976 × 3968 and 3456 × 5184, respectively. The device setups are detailed in Table 3. Due to the large number of individuals participating in data collection, it was not practical for everyone to use the same device. Therefore, researchers utilised two separate devices. The dataset's complexity and value are both enhanced by the use of multiple resolutions. Because it provides models with diverse yet representative inputs, using multiple devices is a popular strategy in this area of study. In practice, operators in the agricultural and non-agricultural sectors use cellphones with different technical specifications, such as resolution.

Table 3: Acquisition device shapes

Technical Parameters	DSRL Camera	Smartphone Camera
Focal ratio	f/4.5	f/2.2
Color space	RGB	RGB
Image size	3456 × 5184	2976 × 3968
Model device	Canon EOS 60D	Honor 6×
Focal length	50 mm	3.83 mm

Throughout the growing season, the leaves were photographed from a variety of angles, under varied lighting conditions (cloudy, sunny, and windy days), against different backgrounds (other plants and weeds), and with varying levels of background noise (Figure 1):

- I was able to capture conditions that could be categorised, which enabled numerous advantages.
- I was able to capture the evolution of visual indications.
- I was able to capture the fruit from the fruit phase.

Figure 1: In the first row, from left to right, you can see pear leaves photographed in various light conditions: dispersed light, intense sunlight reflection, indirect sunlight, direct sunlight and the second row displays photographs of pears at various developmental stages

4. Methodology

4.1. Segmentation Using DBPN

DBPNet keeps the encoder-decoder structure and uses the feature extractor from the pre-trained VGG16 classification model on the ImageNet dataset as the encoder. In the decoding stage, TBSEM is combined with spatial attention in the low-level feature branch [24]. The enhanced ASPP is employed to augment the receptive field of high-level features within the semantic branch, thereby maximizing the features' attributes and benefits while mitigating the influence of detrimental factors on model performance. Figure 2 shows how the projected model [25] is put together.

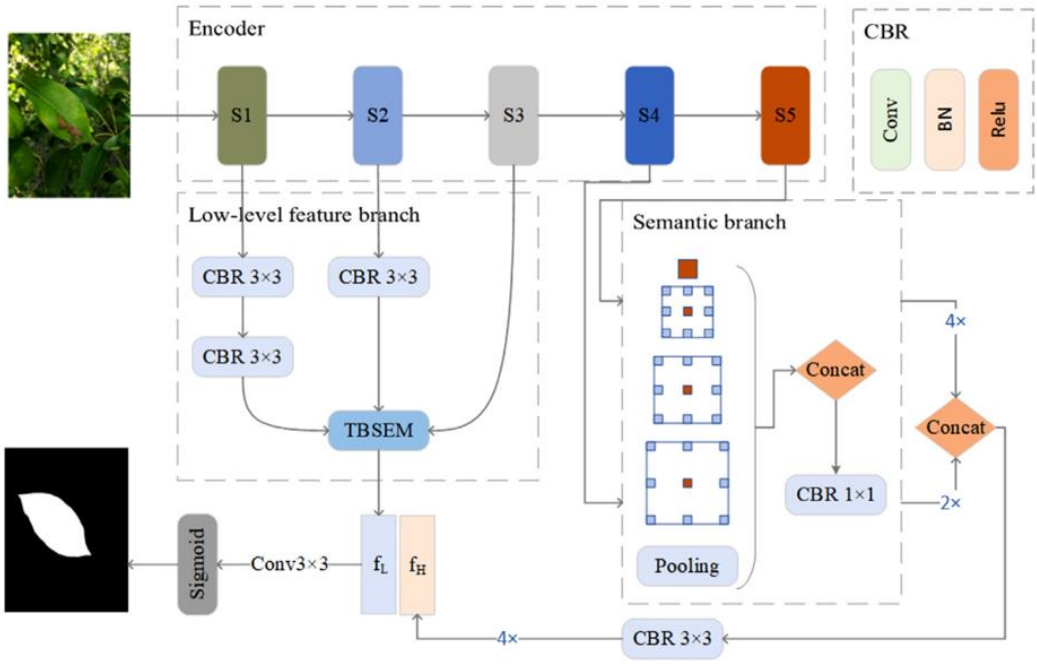

Figure 2: Double-branch polymerisation net

The leaf image has interference pixels, like shadows from lights, branch occlusions, and plastic film, which make its spatial information more complicated. The current research emphasizes the optimization of deep depth features during feature fusion or reuse, neglecting the role of shallow low-level features in the restoration of pear leaf details, and is unable to concentrate on diseased pear leaves in the spatial dimension. To do this, a three-branch space enhancement module (TBSEM) is made to pick up on useful spatial local dependencies in the low-level feature branch and make the model better at shaping details and paying

attention to the diseased pear leaf area. The module takes in S1, S2, and S3, which are three-branch features. The S1–S3 features have fewer convolutional layers, not enough semantic information, and are very similar to each other. Because of this, combining branch features can easily lead to information redundancy. The input features provide useful spatial detail, and the branch features are adaptively fused in the spatial dimension with independent weights (adaptive feature fusion, AFF) S1–S3. When the gradient changes, the model automatically changes the weights of the different feature layers to make the effective ones stand out more. The formula of adaptive fusion is defined as:

$$y_{ij}^{l} = a_{ij}^{l}.x_{ij}^{1\to l} + \beta_{ij}^{l}.x_{ij}^{2\to l} + \gamma_{ij}^{l}.x_{ij}^{3\to l} \quad (1)$$

Where $x_{ij}^{1\to l}, x_{ij}^{2\to l}, x_{ij}^{3\to l}$ represents the shallow features S1,S2,S3, $a_{ij}^{l}, \beta_{ij}^{l}, \gamma_{ij}^{l}$ represents the weights of shallow features, and y_{ij}^{l} represents the fused features. Then, global information and salient features are extracted by two pooling methods. After splicing, multi-scale features are extracted using 3 × 3, 5 × 5, and 7 × 7 convolution kernels, and then 1 × 1 convolutional dimension reduction is applied. The activation function uses Sigmoid to map the features to the (0, 1) interval, producing a two-dimensional attention map, which is then multiplied by the original features to yield the output of the TBSEM module.

4.1.2. Semantic Branch

It was decided to pass S4–S5 to the advanced semantic branch because they were very close to the encoder output, had low resolution, and contained rich semantic information. Because the encoder features have the disadvantages of an insufficient receptive field and a single feature scale, the ASPP module is selected to optimise advanced features, and multi-scale information from stacked receptive fields with different expansion rates is obtained via dilated convolution. When associated with the convolution, the dilated convolution has a unique expansion rate parameter that is significantly different [26]. Based on the standard convolution kernel, r-1 0 is placed between every two elements when the expansion rate is r. This indicates that the expansion rate is determined by itself. It is possible to obtain larger-scale features using the dilated convolution technique, which widens the feature receptive field without introducing additional parameters.

For the purpose of obtaining receptive fields of varying scales, the ASPP is coupled in parallel with dilated convolutions of varying expansion rates in DeepLabV2 [27]. By incorporating global context information, the improved ASPP in DeepLabV3 can counteract the detrimental influence of incorrect information introduced by high-expansion-rate convolution kernels, thereby mitigating the degradation of convolution kernel efficacy. This is accomplished by introducing image-level global features. Additionally, multi-scale features in the channel dimension are fused with the global spatial pooling of the original image, which is coupled in parallel with dilated convolutions.

4.2. Classification Using Modified Swin Transformer

In this part, the planned network architecture is detailed. Figure 3 shows how the model sends the input image to the LPSW backbone network, which stands for Local Perception Swin Transformer. Following the FPN structure, the feature map is sent to the SAIEC network model for spatial attention interleaved execution cascade. Classifications using feature maps and bounding segmentation are handled by the model's back end. Our approach categorises each bounding box as either containing an object or not. Below, you will find an introduction to each module, including complete information.

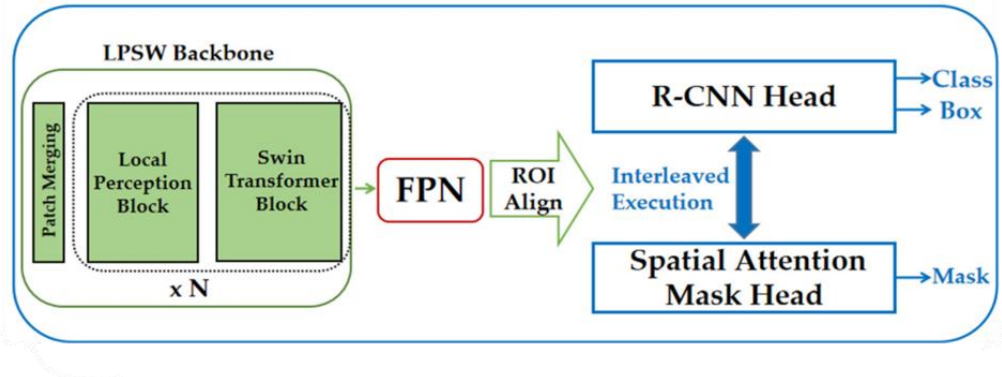

Figure 3: Model flow diagram including Region of Interest (ROI) constructions, which integrate the suggested local perceptual network architecture

4.2.1. Local Perception Swin Transformer (LPSW) Backbone

Swin-T and Swin-L are the four diverse variants of the Swin transformer. These versions are arranged from largest to smallest [28]. This study presents Swin-T, which accounts for the specificity and computational complexity of images from remote sensing observations. Two, two, six, and two blocks are present in each stage, correspondingly. As with ViT, it begins by dividing an RGB image into non-overlapping patches using the patch partition layer. When each patch is applied, it is considered a "token," and its feature is configured to be the RGB values of the raw pixels. Each of the four stages that make up the Swin transformer is responsible for producing a particular quantity of tokens. A token is a 4×4 raw image patch. This vector is given an image with dimensions H × W. To map this token to a vector of dimension C, a linear embedding is now being used. During the first, second, third, and fourth stages, the tokens that are produced are as follows: $H/4 \times W/4, H/8 \times W/8, H/16 \times W/16,$ and $H/32 \times W/32$. At each stage, there is a patch-merging block comprising a partition layer and a linear embedding layer. Additionally, there are some Swin transformer blocks and a local perception block.

4.2.2. Swin Transformer Block

At its heart is the Swin block, which is an algorithmic component. To improve the training stability, researchers insert a layer norm (LN) layer in the centre and apply a residual connection after each module. For this section, researchers have Equation (2):

$$\begin{aligned} \widehat{X}^{l} &= W - MSA\left(LN\left(X^{l-1}\right)\right) + X^{l-1} \\ X^{l} &= MLP\left(LN\left(\widehat{X}^{l}\right)\right) + \widehat{X}^{l} \\ \widehat{X}^{l+1} &= SW - MSA\left(LN\left(X^{l}\right)\right) + X^{l} \\ X^{l+1} &= MLP(LN(\widehat{X}^{l+1}))\widehat{X}^{l+1} \end{aligned} \tag{2}$$

4.2.2.1. W-MSA and SW-MSA

The W-MSA in the Swin transformer block, in contrast to the MSA in the traditional ViT, uses a window as a unit to control the calculation area; the default window size is 7, which simplifies the problem by reducing the amount of network computation to a linear function of the window size. Because MSA does not provide cross-window connections, a new window-segmentation mechanism is required for SW-MSA to succeed where W-MSA failed: enabling communication between windows. The window is segmented according to W-MSA based on these shifts, and SW-MSA differentiates from W-MSA in its window segmentation mechanism.

4.2.2.2. Local Perception Block (LPB)

It is highly unlikely that the site transformer will be able to detect the structural details and local correlation in the image, despite the construction using a shift window scheme consisting of successive layers organised in a hierarchical construction; a significant amount of information is still not adequately recorded. Researchers proposed a local perception block (LPB) preceding the Swin transformer subblock to address this issue. The first step the LPB takes is to transform a collection of vector features into a 3-D feature map. This is done because the data Swin operates on are vectors rather than feature maps, unlike conventional CNNs. An example of this would be the transformation of a token (B, H * W, C) into a feature map (B, C, H, W). Following the addition of a 3×3 convolution with a GELU activation and a dilation of 2, a residual connection is utilised to enhance the extraction of local spatial characteristics while maintaining an appropriately large receptive field. The final step involves reshaping the feature map to the form (B, H, W, C) and then sending it to the block. As a result of the properties of the dilated field of images, the field is expanded, enabling a wide variety of contextual information to be encoded effectively at multiple scales. When compared to the conventional convolution procedure, the dilated convolution method allows for the enlargement of the receptive field. Notably, the conventional convolutions with dimensions 3 3 each have a field that is also 3 × 3. Considering that it is a dilated convolution with a dilation of two and a kernel size of seven by seven, the receptive field is seven by seven. As a result, dilated convolution can enlarge the field of influence without compromising feature resolution.

4.2.3. Spatial Attention Interleaved Execution Cascade (SAIEC)

The definition of samples at various stages is the primary focus of the Cascade R-CNN proposal. As a result of differences across stages, the detector gives greater weight to the positive models that fall below the threshold. Compared to the input IoU threshold, the output IoU threshold is higher, leading to more positive samples in the subsequent stage. The detector effect can gradually become more effective as the relationship between steps grows. The Cascade Mask R-CNN is a straightforward implementation of the Cascade R-CNN architecture. Although it significantly improves in mask AP. Cascade Mask R-CNN is

improved in this study, and a novel framework for instance segmentation, called the spatial attention inserted execution cascade (SAIEC), is proposed. This was done since the HTC technique served as the inspiration for this study. The precise strategies for improvement are as described below.

4.2.3.1. Interleaved Implementation and Mask Information Flow

As shown in Figure 4, the research led to improvements in the CNN. Although Cascade R-CNN incorporates communication between the two branches throughout training, this communication occurs in parallel. Consequently, researchers suggest interleaved execution, in which the box branch is performed first at each stage and then passed to the mask branch to forecast the mask, as demonstrated in Figure 4. In the diagram, the letter F represents the backbone system's characteristics, the letter P represents ROI pooling, and the letters Bi and Mi represent the stages. Not only does this increase contact among the various departments within each stage, but it also closes the gap between procedures.

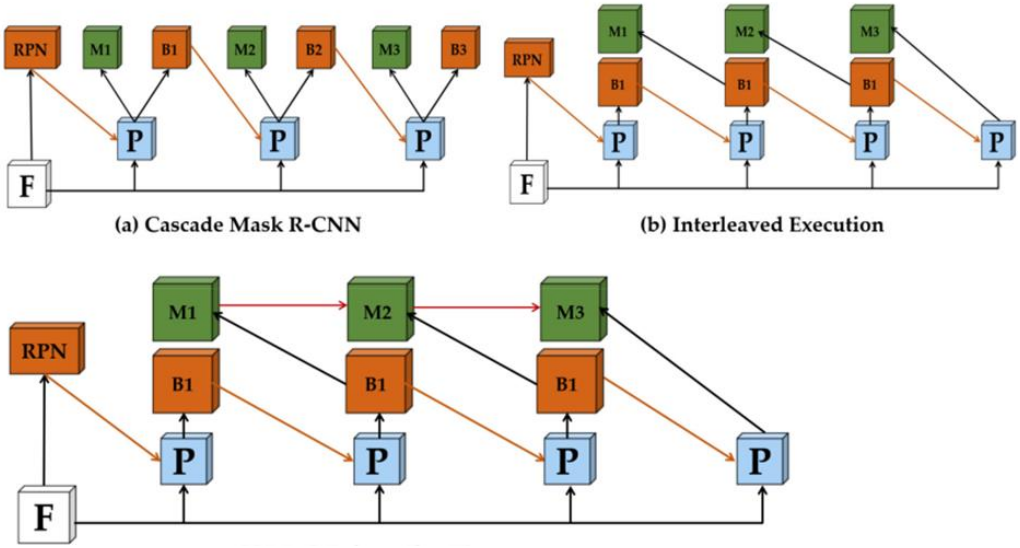

Figure 4: The progress made by the cascade mask head enhancement method, the cascade mask R-CNN system head, the execution in the head, and the final structure of the network head following the addition of mask information flow comprise the network head.

On the other hand, in the Cascade Mask R-CNN, the only branch that affects the subsequent stage is the one currently being processed, and the mask branch connecting to it has no direct information flow. To address this issue, researchers introduced a connection between adjacent mask branches, as depicted in Figure 4. Based on the study's findings, the mask material flow was provided for the branch. This allowed M_(i+1) to acquire the characteristics of M_i. For the investigation, the Mi feature was utilised to perform embedding via a 1×1 convolution. Subsequently, the feature was entered into M_(i+1). By working in this manner, M_(i+1) acquired the features of not only the backbone but also the stage that preceded it.

4.2.3.2. Spatial Attention Mask Head

Using the attention approach, one can focus on key aspects while blocking out unnecessary noise [29]. The spatial mask head was developed by the study, inspired by the spatial attention mechanism [30]. The attention module was used to direct the mask head, which was meant to emphasise pixels that were not relevant to the study. For the study, an enhanced spatial attention module was developed and added before the transposed convolution. Four channels within the head must process the resized local features. In addition, the enhanced spatial attention module must be applied after the local features have been resized. To begin, the enhanced spatial attention module uses average and max pooling to create pooled features P_max and P_avg, respectively. These features are subsequently aggregated through concatenation. The sigmoid function is used to normalise the 3 × 3 dilated convolution layer that follows. What follows is an overview of the calculation procedure:

$$X_{sa} = X_i \otimes \text{sigmoid}(D_{3\times3}(P_{max} \circ P_{avg})) \quad (3)$$

Where $\otimes$ signifies element-wise multiplication, X_{sa} is the attention- map, $D_{3\times3}$ is the 3 × 3 conv layer, and ◦ symbolises the joining process. Then, the category of the particular mask is predicted using a 1 × 1 convolution, and a 2 × 2 deconvolution is

employed for upsampling. Researchers finished the design of the SAIEC framework's mask branch by combining the elements mentioned above. Incorporating a spatial attention mechanism to aid in object concentration and noise reduction, the spatial attention mask head also recovers the network's cross-stage information transfer.

4.2.4. Hyper-Parameter Tuning Using an Improved Beetle Swarm Optimisation Algorithm

There is a strong correlation between the beetle's initial position and the outcomes produced by the BAS algorithm. To put it another way, the initial location that is selected has a significant impact on the effectiveness and efficiency of the optimisation process. In PSO, birds in a flock are simulated by constructing massless particles. Each particle agrees to a fitness value that is defined by a fitness function. PSO is named after the particle swarm optimisation algorithm. A further improvement to the BAS procedure, inspired by the PSO algorithm, extends individuals into groups [31]. That would be the BSO procedure that will be presented. The fundamental idea behind the method is to employ the beetle as a substitute for the particles used in the swarm procedure. In other words, the BAS optimisation technique replaces the optimal values used in the particle swarm algorithm. In a particle swarm optimisation, the initial beetles are identical to the object being optimised. The manner in which the beetle's position is updated during the iterative process is not only dependent on the solution now being utilised by the individual beetle, but also on each succeeding iteration, where concentration. $X = (X_1, X_2, .., X_n)$ is used to space, where $X_i = (x_{i1}, x_{i2}, \dots, x_{iS})^T$ is an S-dimensional vector that shows where beetle i is in the search space and could be a solution to the optimisation problem.$V_i = (v_{i1}, v_{i2}, \dots, v_{iS})^T$ is beetle i. The separate extreme is represented by $P_i = (p_{i1}, p_{i2}, \dots, p_{iS})^T$, and the global danger is characterised by $P_g = (p_{g1}, p_{g2}, \dots, p_{gS})^T$. The speed and site updates of the BSO procedure can be uttered as shadows:

$$x_{is}^{k+1} = x_{is}^{k} + \lambda v_{is}^{k} + (1-\lambda)\xi_{is}^{k} \quad (4)$$

$$v_{is}^{k+1} = \omega v_{is}^{k} + c_1 r_1 \left(p_{is}^{k} - x_{is}^{k}\right) + c_2 r_2 \left(p_{gs}^{k} - x_{gs}^{k}\right) \quad (5)$$

$$\xi_{is}^{k+1} = \delta^{k} * v_{is}^{k} * \operatorname{sign}\left(f\left(x_{rs}^{k}\right) - f\left(x_{ls}^{k}\right)\right) \quad (6)$$

$$x_{ls}^{k+1} = x_{rs}^{k} + v_{is}^{k} * \frac{\vec{d}}{2}; x_{ls}^{k+1} = x_{ls}^{k} - v_{is}^{k} * \frac{\vec{d}}{2} \quad (7)$$

Where $s = 1,2, \dots . S; i = 1,2, \dots, n$ and k is the current sum of repetitions. V_{is} is expressed as the speed of the beetles, and ξ_{is} characterises the upsurge in the beetle site drive. The loosening factor (λ) and inertia weight (ω) are limits, and r_1, r_2 are random of (0; 1). The parameters c_1 and c_2 regulate the impression degree of the different beetles. Specifically, as inertial weight ω is increased in the BSO algorithm, the search range that corresponds to the particle expands. This implies that the algorithm has stronger global search but weaker local search. When the weight ω is smaller, the search range that corresponds to the atom is also higher. This means that the particle narrow has a stronger ability to search locally, while its ability to search globally is weaker. Given the goal of enhancing the inertia weight, this article proposes modifications to the BSO. Regarding the inertia weight formula, the following modifications are made in this paper:

$$\omega(k) = \text{rand} * \omega_{\min} * (1 - \cosh) + \omega_{\max} * \cosh \quad (8)$$

Where rand is a random variable (0, 1), $\omega_{\min} = 0.4, \omega_{\max} = 0.9, h = \pi t/2k_{\max}$; and $k_{\max}$ is the maximum sum of repetitions during iteration. During the initial phase of the search, the enhanced strategy exhibits greater value and a slower rate of σ, which is advantageous for the algorithm in extending the duration of the global search. The likelihood of discovering an optimal global-scale solution has increased. By utilising the fast ω, the capability to continuously track the solution throughout the later stages of the search is enhanced, leading to the discovery of the global optimal key and improved system accuracy. The upgraded BAS procedure is merged with the PSO procedure to obtain the IBSO. This is done in accordance with the development factor in the BAS procedure. When the IBSO procedure is iterated, the site update works in conjunction with the apparatus to learn about update techniques that could accelerate overall progress and reduce the likelihood that the population will converge on an optimal solution given its local conditions. Not only does the IBSO algorithm belong to the global optimisation category, but it also possesses possibilities for exploration and development. Moreover, the linear grouping enhances the speed and accuracy of population optimisation, thereby making the method more stable.

5. Results and Discussion

As the hardware platform for the experiment, the research utilised a computer equipped with a GeForce RTX 3060 GPU (12 GB) throughout the experiment. Python 3.8 and Python 1.8.1 were the versions of Python used in the compilation environment, and researchers used Python as the DL framework. Figure 5 illustrates the loss of learning rate and the batches utilised in the proposed model.

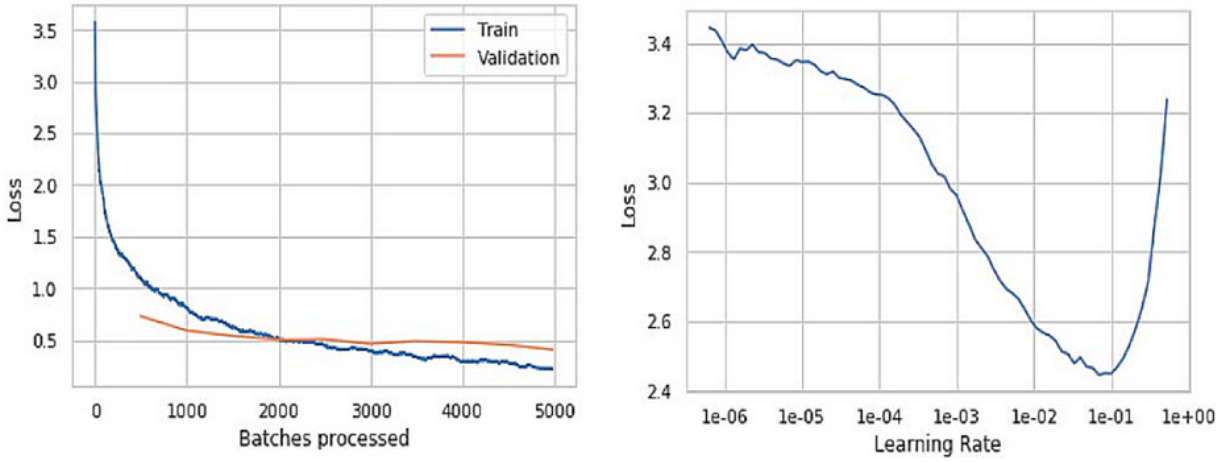

Figure 5: The training, validation, and learning rate of the final model

5.1. Experimental Analysis of Proposed Model

In Table 4, characterize the trial consequences related to OA using the F1 score. In the Accuracy (max) analysis, the ELM model achieved 0.8874, and the CNN model achieved 0.8940. The RNN model attained 0.7129, the LSTM model attained 0.8906, and the proposed model attained 0.9141. Then the Accuracy (avg) was 0.8839, the CNN model reached 0.8579, the RNN model reached 0.7078, the LSTM model reached 0.8763, and the lastly proposed model reached 0.9026. Then, the ELM, CNN, RNN, LSTM, and proposed models all achieved 20k in accuracy. Then the Epoch of ELM model attained 2, the CNN model attained 13, the RNN model attained 13, the LSTM model attained 8, and the lastly proposed model attained 12. Then, the F1 scores were 0.8875 for the ELM model, 0.8816 for the CNN model, 0.7708 for the RNN model, 0.8887 for the LSTM model, and 0.9018 for the proposed model.

Table 4: According to the optimal accuracy score (F1), the results of the experiment

Model	ELM	CNN	RNN	LSTM	Proposed
Precision	0.8828	0.8281	0.7081	0.8882	0.8795
Data capacity	20 k	20 k	20 k	20 k	20 k
Accuracy(max)	0.8874	0.8940	0.7139	0.8916	0.9141
Accuracy(avg)	0.8939	0.8589	0.7078	0.8773	0.9126
Data capacity	20 k	20 k	20 k	20 k	20 k
Epoch	2	13	13	8	12
F1 Score	0.8875	0.8826	0.7708	0.8887	0.9028
Recall	0.8947	0.9514	0.8562	0.8929	0.9157
Epoch	2	4	5	23	10

Then the Recall of the ELM model was 0.8947, the CNN model was 0.9504, the RNN model was 0.8562, the LSTM model was 0.8919, and the lastly proposed model attained 0.9057. Then the Precision of the ELM model was 0.8818, the CNN model was 0.8281, the RNN model was 0.7081, the LSTM model was 0.8872, and the lastly proposed model attained 0.8975. Then, the ELM, CNN, RNN, LSTM, and proposed models all achieved 20k in accuracy. Then the Epoch of ELM model attained 2, the CNN model attained 4, the LSTM model attained 5, the RNN model attained 23, and the lastly proposed model attained 10 in accuracy.

Table 5: Experimental consequences leading to optimal data capacity

Model	ELM	Proposed	CNN	RNN	LSTM
Accuracy(max)	0.87	0.902	0.825	0.726	0.879
Precision	0.868	0.901	0.830	0.696	0.874
Data capacity	11 k	13 k	7 k	7.5 k	12.5 k
Accuracy(avg)	0.86	0.874	0.805	0.711	0.855
Data capacity	9.5 k	13 k	6.5 k	14 k	12.5 k
Epoch	10	10	10	10	10

F1 Score	0.87	0.901	0.813	0.694	0.877
Recall	0.88	0.905	0.799	0.890	0.886
Epoch	10	10	10	10	10

In Table 5, characterise the experimental consequences that yield optimal data volume. In the analysis of Accuracy(max), the ELM model attained 0.871, the proposed model attained 0.902, the CNN model attained 0.825, the RNN model attained 0.726, and the LSTM model attained the optimal data capacity of 0.879. Then the CPrecisionC0.868, and then the RNN model attained 0.901, and the CNN model attained 0.830; the RNN model attained 0.696, and the LSTM model attained the optimal data capacity of 0.874. Then the ELM model volume reached 11 k, the proposed model reached 13 k, the RNN model reached 7 k, the RNN model reached 7.5 k, and the LSTM model reached the optimal data capacity of 12.5 k. Then the Accuracy (avg) of the ELM model was 0.862, the proposed model was 0.874, the CNN model was 0.805, the RNN model was 0.711, and the LSTM model was 0.855, corresponding to the optimal data capacity.

Then, the ELM model attained 9.5 k, the proposed model attained 13 k, the CNN model attained 6.5 k, the RNN model attained 14 k, and the LSTM model attained the optimal data capacity of 12.5 k. Then the Epoch of ELM model attained 10, the proposed model attained 10, the CNN model attained 10, the RNN model attained 10, and the LSTM model attained the optimal data capacity of 10. Then, the F1 scores were 0.875 for the ELM model, 0.901 for the proposed model, 0.813 for the CNN model, and 0.694 for the RNN model. Lastly, the LSTM model attained the optimal data capacity of 0.877. Then the Recall of the ELM model was 0.887, the proposed model was 0.905, the CNN model was 0.799, the RNN model was 0.890, and the LSTM model attained the optimal data capacity of 0.886. The ELM model attained 10, the proposed model attained 10, the CNN model attained 10, the RNN model attained 10, and the LSTM model attained the optimal data capacity of 10.

Table 6: Overall accuracy (OA) and training time were cut by five times for the same model that was taught in five different batch sizes

Batch Size	OA	Training Time
8	93.54 ± 0.67	3 h 45 min
16	93.65 ± 0.73	1 h 56 min
32	93.59 ± 0.74	1 h 06 min
64	93.43 ± 0.71	34 min
128	93.45 ± 0.83	19 min

According to the data presented in Table 6, the training time and overall accuracy (OA) have increased by a factor of 5 for the same pattern that was learned across five different batch sizes. With a batch size of eight, the average overall accuracy was 93.54 ± 0.67, and the training time was 3 hours and 45 minutes. With a batch size of 16, the average overall accuracy was 93.65 ± 0.73, and the training time was 1 hour and 56 minutes. For a batch size of 32, the average overall accuracy was 93.59 ± 0.74, and the training time was 1 hour and 6 minutes. After that, a 64-batch size yielded an average overall accuracy of 93.43 ± 0.71 and a training time of 34 minutes. Following that, a batch size of 128 was selected, yielding an average overall accuracy of 93.45 ± 0.83, and the training time was 19 minutes.

6. Conclusion and Future Work

Based on a deep neural network, a new framework is proposed for categorising pear leaf disease in a complex background. The experiments demonstrate that the proposed method is effective for segmenting pear leaves and classifying diseases that affect them. Our conclusion that the DBPN enables high-precision leaf extraction while preserving whole-edge lesion information to assist in high-precision lesion classification is supported by the findings. The purpose of this paper was to develop the Swin transformer for the classification of input segmented images by analysing the benefits and drawbacks of both transformers and CNNs. Additionally, researchers designed the (LPSW) backbone network. In conclusion, the suggested model developed the (SAIEC) network framework with the intention of enhancing the accuracy of mask prediction for classification tasks.

The novel combination of CNNs and transformers, which leverages the strengths of both local and global information, has the potential to dramatically improve object detection accuracy. Incorporating the interweaved execution construction and the enhanced unit into the mask head can reduce noise and improve the network's ability to predict the mask. On the other hand, due to the limited availability of public datasets and photos of pear leaf diseases, research on the topic focuses on only three illnesses: pear leaf black spot, pear leaf rust, and pear leaf slug. Additionally, the number of diseases that can be identified is rather low. It is possible that in the future, the research may concentrate on broadening the scope of the data sets and disease classifications.

Acknowledgement: The authors would like to express their sincere gratitude to St. Joseph's College of Engineering, New Horizon College of Engineering, Quest Technologies, and Sunlife Canada Financials for their academic and technical support. Their valuable guidance and resources greatly contributed to the successful completion of this research.

Data Availability Statement: The dataset supporting the findings of this study is available from the corresponding author upon reasonable request, in accordance with transparency and reproducibility guidelines.

Funding Statement: The authors confirm that this research and the preparation of the manuscript were carried out without any external funding support from agencies or organisations.

Conflicts of Interest Statement: The authors collectively declare that there are no competing interests or personal relationships that could have influenced the outcomes of this study.

Ethics and Consent Statement: All authors have jointly agreed to the publication of this work and affirm that it adheres to ethical standards, with consent granted for open access and academic use by the research community.

References

1. W. Alosaimi, H. Alyami, and M. I. Uddin, "PeachNet: Peach diseases detection for automatic harvesting," *Computer Modeling in Engineering and Sciences*, vol. 67, no. 2, pp.1665-1677, 2021.
2. N. Yao, F. Ni, Z. Wang, J. Luo, W. K. Sung, C. Luo, and G. Li, "L2MXception: An improved Xception network for classification of peach diseases," *Plant Methods*, vol. 17, no. 1, p. 36, 2021.
3. Y. Li, A. Li, X. Li, and D. Liang, "Detection and identification of peach leaf diseases based on YOLOv5 improved model," in *Proc. 5th Int. Conf. Control and Computer Vision (ICCCV)*, New York, United States of America, 2022.
4. M. Akbar, M. Ullah, B. Shah, R. U. Khan, T. Hussain, F. Ali, F. Alenezi, I. Syed, and K. S. Kwak, "An effective deep learning approach for the classification of bacteriosis in peach leave," *Frontiers in Plant Science*, vol. 13, no. 11, p. 1064854, 2022.
5. H. Farman, J. Ahmad, B. Jan, Y. Shahzad, M. Abdullah, and A. Ullah, "EfficientNet-based robust recognition of peach plant diseases in field images," *Computers, Materials and Continua*, vol. 71, no. 1, pp. 2073–2089, 2022.
6. M. I. A. Abenina, J. M. Maja, M. Cutulle, J. C. Melgar, and H. Liu, "Prediction of potassium in peach leaves using hyperspectral imaging and multivariate analysis," *AgriEngineering*, vol. 4, no. 2, pp. 400–413, 2022.
7. M. H. Saleem, J. Potgieter, and K. M. Arif, "A performance-optimized deep learning-based plant disease detection approach for horticultural crops of New Zealand," *IEEE Access*, vol. 10, no. 8, pp. 89798–89822, 2022.
8. N. Yao, F. Ni, M. Wu, H. Wang, G. Li, and W. K. Sung, "Deep learning-based segmentation of peach diseases using convolutional neural network," *Frontiers in Plant Science*, vol. 13, no. 5, p. 876357, 2022.
9. J. Andrew, J. Eunice, D. E. Popescu, M. K. Chowdary, and J. Hemanth, "Deep learning-based leaf disease detection in crops using images for agricultural applications," *Agronomy*, vol. 12, no. 10, p. 2395, 2022.
10. Q. Li, W. Sun, A. Shi, C. Lei, and S. Mu, "Image detection of peach diseases and pests," Proc. Int. Conf. Image, Vision and Intelligent Systems (ICIVIS), *Springer*, Singapore, 2022.
11. P. Dhar, M. S. Rahman, and Z. Abedin, "Classification of leaf disease using global and local features," *International Journal of Information Technology and Computer Science*, vol. 14, no. 1, pp. 43–57, 2022.
12. R. Dwivedi, T. Dutta, and Y. C. Hu, "A leaf disease detection mechanism based on L1-norm minimization extreme learning machine," *IEEE Geoscience and Remote Sensing Letters*, vol. 19, no. 9, pp. 1–5, 2021.
13. T. Arunadevi, K. Aravinda, V. Revathi, and P. K. Balasubramanian, "Designing a modified grey wolf optimizer based CycleGAN model for EEG MI classification in BCI," *SSRN*, 2023. Available: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4642989 [Accessed by 23/06/2024].
14. R. Alguliyev, Y. Imamverdiyev, L. Sukhostat, and R. Bayramov, "Plant disease detection based on a deep model," *Soft Computing*, vol. 25, no. 21, pp. 13229–13242, 2021.
15. P. Bedi and P. Gole, "Plant disease detection using hybrid model based on convolutional autoencoder and convolutional neural network," *Artificial Intelligence in Agriculture*, vol. 5, no. 1, pp. 90–101, 2021.
16. S. Yadav, N. Sengar, A. Singh, A. Singh, and M. K. Dutta, "Identification of disease using deep learning and evaluation of bacteriosis in peach leaf," *Ecological Informatics*, vol. 61, no. 3, p. 101247, 2021.
17. S. Parez, N. Dilshad, N. S. Alghamdi, T. M. Alanazi, and J. W. Lee, "Visual intelligence in precision agriculture: Exploring plant disease detection via efficient vision transformers," *Sensors*, vol. 23, no. 15, p. 6949, 2023.
18. T. U. Rehman, M. Alam, N. Minallah, W. Khan, J. Frnda, S. Mushtaq, and M. Ajmal, "Long short-term memory deep net performance on fused PlanetScope and Sentinel-2 imagery for detection of agricultural crop," *PLOS ONE*, vol. 18, no. 2, p. e0271897, 2023.

19. R. Punithavathi, A. D. C. Rani, K. R. Sughashini, C. Kurangi, M. Nirmala, H. F. T. Ahmed, and S. P. Balamurugan, "Computer vision and deep learning-enabled weed detection model for precision agriculture," *Computer Systems Science and Engineering*, vol. 44, no. 3, pp. 2759–2774, 2023.
20. Q. Wang, D. Wu, Z. Sun, M. Zhou, D. Cui, L. Xie, D. Hu, X. Rao, H. Jiang, and Y. Ying, "Design, integration, and evaluation of a robotic peach packaging system based on deep learning," *Computers and Electronics in Agriculture*, vol. 211, no. 8, p. 108013, 2023.
21. Y. Wang, X. Jin, J. Zheng, X. Zhang, X. Wang, X. He, and M. Polovka, "An energy-efficient classification system for peach ripeness using YOLOv4 and flexible piezoelectric sensor," *Computers and Electronics in Agriculture*, vol. 210, no. 7, p. 107909, 2023.
22. P. Teng, Y. Zhang, T. Yamane, M. Kogoshi, T. Yoshida, T. Ota, and J. Nakagawa, "Accuracy evaluation and branch detection method of 3D modeling using backpack 3D LiDAR SLAM and UAV-SfM for peach trees during the pruning period in winter," *Remote Sensing*, vol. 15, no. 2, p. 408, 2023.
23. G. Fenu and F. M. Malloci, "Using multioutput learning to diagnose plant disease and stress severity," *Complexity*, vol. 2021, no. 1, p. 6663442, 2021.
24. K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv Preprint*, 2014. Available: https://arxiv.org/abs/1409.1556 [Accessed by 04/06/2024].
25. L. C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. ECCV*, 2018. Available: https://openaccess.thecvf.com/content_ECCV_2018/papers/Liang-Chieh_Chen_Encoder Decoder_with_Atrous_ECCV_2018_paper.pdf [Accessed by 07/06/2024].
26. F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," *arXiv Preprint*, 2015. Available: https://arxiv.org/abs/1511.07122 [Accessed by 23/06/2024].
27. L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, 2017.
28. Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin Transformer: Hierarchical vision transformer using shifted windows," *arXiv Preprint*, 2021. Available: https://arxiv.org/abs/2103.14030 [Accessed by 25/06/2024].
29. S. Woo, J. Park, J. Y. Lee, and I. Kweon, "CBAM: Convolutional block attention module," *arXiv Preprint*, 2018. Available: https://arxiv.org/abs/1807.06521 [Accessed by 17/06/2024].
30. X. Zhu, D. Cheng, Z. Zhang, S. Lin, and J. Dai, "An empirical study of spatial attention mechanisms in deep networks," *arXiv Preprint*, 2019. Available: https://arxiv.org/abs/1904.05873 [Accessed by 11/06/2024].
31. T. Wang and L. Yang, "Beetle swarm optimization algorithm: Theory and application," *arXiv Preprint*, 2018. Available: https://arxiv.org/abs/1808.00206 [Accessed by 01/06/2024].
32. G. Fenu and F. M. Malloci, "DiaMOS Plant Dataset: A Dataset for Diagnosis and Monitoring Plant Disease," *Zenodo*, 2021. Available: https://zenodo.org/records/5557313 [Accessed by 11/06/2024].