International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Deepharvest: Revolutionizing Agriculture Through a Variofusionnet and Featexpronet for Accurate and Timely Leaf Disease Detection and Management

[1*]M. Chithambarathanu, [2]Dr. M. K. Jeyakumar

Submitted: 29/01/2024 **Revised**: 07/03/2024 **Accepted**: 15/03/2024

Abstract: In this pioneering approach to rust classification in plant leaves, deploy an exhaustive pre-processing pipeline to fortify the robustness of this dataset. The integration of Generative Adversarial Networks serves to augment the dataset, while a groundbreaking Modified Gaussian Smoothing technique is introduced to effectively mitigate noise and elevate image quality. Feature extraction is bolstered through Contrast Stretching, enhancing contrast, and color correction methods adeptly standardize color variations. Precision in disease-affected area identification is achieved through refined leaf localization using Region Proposal Networks (RPN) and this innovative Spatial Attention Mechanisms. Further optimization in Regions of Interest (ROI) identification is realized with an optimized dual attention YOLO and FeatExProNet combination, extracting key features encompassing shape, color, texture, statistics, and deep learning-based attributes. Feature selection employs a Hybrid Optimization Approach, synergizing Binary Sand Cat Swarm Optimization and Butterfly Optimization algorithms. The conclusive step incorporates a VarioFusionNet-based model, seamlessly amalgamating Vision Transformer, Google Net, Alex Net, DenseNet-121, ResNet-50, and Efficient Net to ensure unparalleled accuracy in leaf disease detection. This comprehensive methodology represents a remarkable leap forward in rust classification, offering a commitment to improved accuracy and robustness in the identification of plant leaf diseases.

Keywords: *Alex Net, DenseNet-121, Efficient Net, Generative Adversarial Networks, Google Net, Modified Gaussian Smoothing technique, Region Proposal Networks, ResNet-50, Vision Transformer, YOLO and FeatExProNet.*

1. Introduction

Ensuring the accurate and timely detection of leaf diseases is paramount for effective plant health management. Our innovative approach integrates advanced detection methods to swiftly pinpoint and address issues, contributing to optimal plant well-being [1]. Notably, common apple leaf diseases, including Alternaria leaf spot, Brown spot, Mosaic, Grey spot, and Rust, pose significant threats to yield. The development of a precise and swift detector is imperative for the overall health of the industry [1]. Addressing the impact of tea leaf diseases on yield and quality, this research introduces a low-shot learning method for timely identification and control, enhancing disease spot segmentation through the extraction of color and texture features [2]. For precise apple leaf disease identification, a novel approach utilizes deep convolutional neural networks with a unique Alex Net-based architecture, involving the generation of abundant pathological images [3]. In contrast to traditional visual methods, optical sensors offer non-invasive measurement of pathogen-induced plant physiology changes, proving valuable for disease detection, identification, and quantification across various scales [4]. The development of an automatic identification method for grape leaf diseases is urgent for maintaining grape yield. Taking inspiration from the success of deep learning, this research applies the methodology to grape disease identification [5].

Presenting a groundbreaking model for plant disease recognition through leaf image classification, this approach leverages deep convolutional networks [6]. Innovative training methods facilitate seamless real-world implementation. Additionally, an enhanced Faster R-CNN architecture, with adjusted CNN model parameters, is introduced for the automatic detection of leaf spot disease in sugar beet [7]. Enhancing cucumber leaf spot disease extraction in intricate backgrounds, a refined fuzzy C-means algorithm is introduced for improved accuracy [8]. In agricultural information, automated identification and diagnosis of maize leaf diseases are crucial. Enhanced Google Net and Cifar10 models in deep learning improve accuracy with fewer parameters [9]. Given their superior computational and accuracy capabilities, computer vision and deep learning are favoured for diverse fungal disease classifications. Here, we propose a Multilayer CNN for Mango leaves infected with Anthracnose [10].

Implementing deep convolutional-neural-network models for plant disease identification, replace standard convolution with depth-separable convolution, reducing parameters and computation costs significantly [11]. Enhancing plant disease identification, GPDCNN combines dilated

[1*]*Research Scholar, Department of Computer Science and Engineering, Noorul Islam Centre for Higher Education, Tamilnadu, India.*
[2] *Professor, Department of Computer Applications, Noorul Islam Centre for Higher Education, Tamilnadu, India.*
[1*]*Corresponding Author Email:chithambaramthanu@gmail.com*

convolution with global pooling, increasing receptive fields without complexity, employing dilated convolution for spatial resolution, and integrating their merits [12]. Examined the capability of Sentinel-2 band settings in discerning CLR infection levels in leaves by aligning field spectra. Implemented random forest and PLS-DA algorithms with and without variable optimization [13]. Proposed a Deep Convolutional Neural Network for symptom-specific recognition of four cucumber diseases, incorporating data augmentation to mitigate overfitting [14]. Rice plant leaf images, depicting normal and diseased states, are directly acquired from the field. Pre-processing involves converting RGB to HSV, enabling background removal and segmentation using a clustering method [15].

To achieve a comprehensive advancement in leaf disease detection, this research

proposes the following objectives:

- To implement a modified Gaussian smoothing technique for noise reduction.
- To integrate Region Proposal Networks (RPN) with spatial attention mechanisms for precise leaf localization.
- To develop a FeatExProNet model to extract shape, color, texture, statistical, and deep learning-based features
- To propose a hybrid optimization approach combining Binary Sand Cat Swarm Optimization and Butterfly Optimization algorithms for effective feature selection.
- To formulate VarioFusionNet by integrating vision transformer, Google Net, Alex Net, DenseNet-121, ResNe-50, and Efficient Net for a robust leaf disease detection framework.

This research is organized into distinct sections, commencing with a comprehensive introduction in Section 1 and followed by an extensive literature review in Section 2. Section 3 meticulously outlines the proposed methodology, while Section 4 encapsulates results and discussions. The conclusive section succinctly summarizes the research findings, offering a coherent closure to the research.

2. Literature Review

In 2018, Barbedo *et al*. [16] investigated the pivotal factors that influenced the design and efficacy of deep neural networks in plant pathology. A comprehensive analysis was undertaken, highlighting both advantages and limitations. The arguments were substantiated by literature studies and experiments utilizing an image database. Practical conclusions were drawn, aligning with realistic conditions.

In 2020, Panigrahi *et al.* [17] centered on supervised machine learning techniques, such as Naive Bayes, Decision Tree, K-Nearest Neighbor, Support Vector Machine, and Random Forest, for detecting maize plant diseases from images. The classification methods were scrutinized and compared, revealing that the Random Forest algorithm achieved the highest accuracy at 79.23%. The trained models were designed for farmers to facilitate early disease detection and classification as a preventive measure.

In 2019, Pantazi *et al.* [18] demonstrated an automated approach for identifying crop diseases in a range of leaf sample images from various crop species. Feature extraction employed Local Binary Patterns, and One Class Classification was applied for categorization. The methodology included a dedicated One Class Classifier for each plant health condition, covering healthy, downy mildew, powdery mildew, and black rot.

In 2017, Singh *et al.* [19] underscored the benefits of automating plant disease detection, streamlining monitoring in extensive crop farms and facilitating early symptom identification. An image segmentation algorithm was introduced for automatic detection and classification of plant leaf diseases. A survey on diverse disease classification techniques was conducted, and the pivotal task of image segmentation for plant leaf disease detection was successfully executed using genetic algorithms.

In 2021, Huitron *et al.* [20] featured the training and evaluation of four modern Convolutional Neural Network models for the classification of tomato leaf diseases. Utilizing a subset of 18,160 RGB images from the Plant Village dataset categorized into ten classes, transfer learning was applied. The selected models incorporated a depth-wise separable convolution architecture, ideal for low-power devices. Quantitative and qualitative evaluations were performed, employing quality metrics and saliency maps.

In 2019, Dhingra *et al.* [21] utilized an innovative fuzzy set extended form of neutrosophic logic for segmentation, assessing regions of interest. The resultant neutrosophic image comprised three membership elements: true, false, and intermediate. Leveraging segmented regions, a novel feature subset incorporating texture, color, histogram, and disease sequence regions was evaluated for distinguishing between diseased and healthy leaves. Nine classifiers were employed to showcase the discriminatory power of combined features, with random forest emerging as the dominant technique.

In 2019, Sibiya *et al.* [22] applied convolutional neural network principles to model an image recognition and classification network for identifying maize leaf diseases. Neuroph was employed to train the CNN network using images captured through a smartphone camera. A unique training approach and methodology were implemented, ensuring a rapid and practical system deployment. The developed model successfully recognized and classified

three distinct types of maize leaf diseases alongside healthy leaves.

In 2023, Wu *et al*. [23] centered on harnessing hyperspectral imaging with spectral features, vegetation indices, and textural features for early Gray Mold detection. Hyperspectral images were collected, processed, and competitive adaptive reweighted sampling selected optimal wavelengths. Machine learning models were developed using the selected features, ensuring effective recognition of Gray Mold.

In 2019, Geetharamani *et al*. [24] proposed an innovative plant leaf disease identification model, relying on a Deep Convolutional Neural Network. The model underwent training with a diverse dataset containing 39 plant leaf classes and background images. Employing six data augmentation techniques, including image flipping and gamma correction, enhanced the model's performance, underscoring the efficacy of data augmentation in the training process.

In 2017, Bajwa *et al*. [25] aimed to correlate leaf reflectance with crop disease conditions and identify discriminative wavebands. A microplot experiment gathered data, including 800 leaf spectra, chlorophyll content, and disease ratings for soybean cultivars under various disease treatments. Disease discrimination capability was assessed using vegetation indices, and wavebands were identified through stepwise linear discriminant analysis, logistic discriminant analysis, and linear correlation analysis. The findings were utilized to develop a classification function for identifying plant disease conditions.

2.1 Problem Statement

Table 1: Aim and limitations of the previous research

Author	Method	Aim	Limitation
Dhingra *et al.* [21]	• Innovated computer vision for precise leaf disease identification and classification.	• Enhancing precision in leaf disease identification using innovative computer vision techniques.	• The approach's effectiveness may be influenced by varied environmental conditions impacting image quality.
Sibiya *et al*. [22]	• Employed CNN for accurate maize leaf disease recognition and classification, distinguishing them from healthy leaves	• Enhancing precision in maize leaf disease identification through advanced computational procedures.	• The model's accuracy may vary under diverse environmental conditions and different stages of disease progression.
Wu *et al*. [23]	• Fused hyperspectral imaging, spectral features, vegetation indices for early strawberry disease.	• Achieving early and precise strawberry leaf disease identification through advanced hyperspectral imaging techniques.	• Sensitivity to environmental variations may affect the method's accuracy in different conditions.
Bajwa *et al*. [25]	• Monitored soybean diseases using leaf reflectance.	• Achieving effective soybean disease monitoring through spectral analysis.	• Susceptible to environmental factors impacting reflectance patterns and disease manifestations.

3. Proposed Methodology

The proposed methodology encompasses a structured approach for the meticulous execution of objectives, specifically tailored for comprehensive leaf disease detection. Initiating with the critical phase of pre-processing, advanced techniques are applied to optimize dataset robustness. Generative Adversarial Networks are strategically utilized for image augmentation, enhancing the model's ability to generalize across diverse instances. Additionally, a novel Modified Gaussian Smoothing technique is proposed for noise reduction and improved image quality. Contrast Stretching is employed to enhance image contrast, facilitating improved feature extraction, while color correction methods standardize color variations across images. Moving forward, the methodology integrates advanced techniques for precise leaf localization in the

second step. The combination of Region Proposal Networks and proposed Spatial Attention Mechanisms ensures heightened accuracy in localizing disease-affected areas. The third step involves Region of Interest identification, wherein an Optimized dual attention YOLO model, supplemented by FeatExProNet, extracts a comprehensive set of features. These features encompass shape, color, texture, statistical characteristics, and deep learning-based features using Inception V3, ensuring a holistic representation of leaf characteristics.

Subsequently, the fourth step employs a Hybrid Optimization Approach, unifying Binary Sand Cat Swarm Optimization and Butterfly Optimization algorithms for effective feature selection. This strategic selection optimizes the model's performance by focusing on the most informative features. The final step integrates a VarioFusionNet-based model for leaf disease detection, combining Vision Transformer, Google Net, Alex Net, DenseNet-121, ResNet-50, and Efficient Net architectures. The amalgamation of these advanced models ensures a synergistic approach, enhancing the overall accuracy and effectiveness of the leaf disease detection system. Validation on diverse datasets throughout the methodology guarantees the robustness and adaptability of the proposed approach across various scenarios and conditions.

The methodology will encompass the execution of the outlined objectives is shown in Figure 1, starting with pre-processing techniques, followed by leaf localization, ROI identification, feature extraction, feature selection, architecture integration, and validation on diverse datasets. Each step will be meticulously executed, with a focus on leveraging advanced deep learning techniques.

***Step 1: Pre-processing**:*

- Image Augmentation: Utilize Generative Adversarial Networks (GANs) to augment the dataset, enhancing model robustness.
- Modified Gaussian Smoothing: Apply a modified Gaussian smoothing technique (proposed) for noise reduction and improved image quality.
- Contrast Stretching: Enhance image contrast through contrast stretching, aiding in better feature extraction.
- Color Correction: Implement color correction methods to standardize color variations across images.

Step 2: Leaf Localization:

- Region Proposal Networks (RPN) and Spatial Attention Mechanisms (proposed): Integrate RPN and spatial attention mechanisms to improve the accuracy of leaf localization, ensuring precise identification of disease-affected areas.

***Step 3: ROI identification**:*

- Optimized dual attention YOLO

***FeatExProNet based Feature Extraction**:*

- Shape: Extract features such as area, major and minor axis length, perimeter, and solidity.
- Color: Utilize histogram-based features, capturing color distribution patterns.
- Texture: Employ Tamura and Hara lick texture features for enhanced texture representation.
- Statistical Features: Extract moments (mean, skewness, kurtosis) to capture statistical characteristics.
- Deep Learning-based Features: Leverage Inception V3 for automatic and hierarchical feature extraction.

***Step 4: Feature Selection**:*

- Hybrid Optimization Approach: Employ a hybrid optimization approach combining Binary Sand Cat Swarm Optimization and Butterfly Optimization algorithms for effective feature selection.

***Step 5: VarioFusionNet-based leaf disease detection**:*

Vision transformer, google Net, Alex Net, DenseNet-121, ResNe-50, EfficientNet.

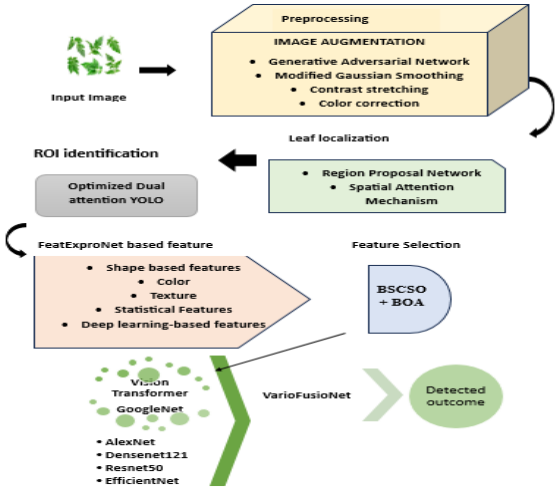


Fig 1: Overall flow diagram of the proposed model

3.1 Reprocessing

In the preprocessing phase, diverse image augmentation techniques are employed, including Generative Adversarial Network integration for realistic data generation. Modified Gaussian Filtering enhances feature extraction, while contrast stretching boosts image clarity. Additionally, meticulous color correction techniques are applied, ensuring optimal input quality for subsequent analyses.

3.1.1 Image augmentation

Data augmentation for images involves creating varied versions of a dataset by manipulating pixel values in a 2-dimensional array. Essentially, it transforms the numerical representation of images, providing a diverse set for improved model training. This process enhances the model's ability to perform and generalize well on a broader range of inputs.

3.1.1.1 Generative Adversarial Network

Generative Adversarial Networks [26] form a methodological class for modelling data distributions, employing a generator (G) to transform random uniform samples into the target distribution and a discriminator (D) to assess sample authenticity. Their joint training, following game-theoretic min-max principles, involves iterative refinement. While GANs excel in visual synthesis, challenges persist, including training difficulties and susceptibility to modal collapse. Ongoing research investigates enhancements such as conditional variables, improved training methods, and task-specific cost functions. Beyond image synthesis, GANs are now being explored in diverse domains like text-to-image synthesis and single-image super-resolution.

3.1.1.2 Modified Gaussian Filtering

Widely adopted in image processing, Gaussian smoothing [27] stands out as a prevalent technique. Described by a 2D Gaussian function featuring a zero mean (μ) and a consistent standard deviation (σ), this method efficiently reduces noise and enhances image quality. The mathematical expression captures the distribution of pixel values, facilitating effective smoothing for diverse applications.

$$G(X,Y) = \exp\left(-\frac{X^2+Y^2}{2\sigma^2}\right) \quad (1)$$

In the context of the Gaussian function in Eq. (1), where $X\ and\ Y$ are variables, the standard deviation significantly influences its behavior. The distribution within $\pm\sigma, \pm 2\sigma, and\ \pm 3\sigma$ encompasses 68%, 95%, and 99.7% of values, respectively. Applied through convolution, the 2D Gaussian function acts as a point spread function, enhancing image quality. Practical implementation involves approximating the Gaussian function discretely, often neglecting values beyond $\pm 3\sigma$ due to their minimal impact, making the convolution kernel practically manageable.

The Gaussian filter serves as a non-uniform low-pass filter, featuring kernel coefficients inversely proportional to distance from the center. The central point holds the highest value, and blurring intensity is determined by the peak width—higher sigma values result in wider peaks. To uphold the Gaussian nature, both kernel size and sigma must increase proportionally. Symmetric and directionally unbiased, the filter's coefficients are sigma-dependent. While computationally efficient due to its separable nature, the Gaussian kernel may not maintain the original image brightness during the filtering process.

For efficient Gaussian function implementation, approximation involves fixing a set number of coefficients, often referred to as the kernel or mask. Optimal smoothing requires a larger sigma (σ), while an accurate representation of the function demands a larger kernel size. A 5 × 5 Gaussian kernel, for instance, is derived by evaluating Eq. (1) across variable ranges of $X\ and\ Y$ from [−2, 2]. This approach balances computational efficiency with the precision needed to capture the desired smoothing effect.

For noise reduction and improved image quality, you can modify the Gaussian smoothing filter by incorporating a weighted average of neighbouring pixels. This modification enhances the filter's ability to reduce noise while preserving important image features. Here's an equation that includes a weighted sum of neighbouring pixels based on a parameter α

In this equation (2):

$$G(X,Y) = \frac{1-\alpha}{\pi-\sigma^2}\ exp\left(\frac{-X^2+Y^2}{2\cdot\sigma^2}\right) + \frac{\alpha}{\pi(2\cdot\sigma)^2}\exp\left(\frac{-X^2+Y^2}{2\cdot(2\cdot\sigma)^2}\right) \quad (2)$$

The first term represents the traditional Gaussian smoothing component.

The second term introduces a weighted sum of neighbouring pixels with a larger standard deviation (here, $2 \cdot \sigma$). The parameter α controls the contribution of this term, determining the balance between noise reduction and blurring. Adjusting α allows you to fine-tune the trade-off between noise reduction and preserving image details. Higher α values result in stronger noise reduction but may lead to more blurring. Experiment with different values of α to find the optimal balance for your specific image processing needs.

3.1.1.3 Contrast stretching

Insufficient illumination, limited dynamic range in image sensors, or misconfigured lens apertures during image acquisition can lead to low-contrast images. Contrast stretching [28] serves to expand the intensity range within an image, ensuring comprehensive coverage of the

recording medium or display device spectrum. The primary aim is to boost image contrast by amplifying darker regions and brightening brighter segments, rejuvenating the overall visual appeal of the image.

$$S = \begin{cases} L * R & 0 \le R < A \\ M * (R - A) + V & A \le R < B \\ N * (R - B) + W & B \le R < l - 1 \end{cases} \quad (3)$$

Incorporating slopes $L, M \text{ and } N$ the contrast stretching transformation, evident in equation (3), selectively darkens dark gray levels (with slopes less than one) and brightens bright gray levels (with slopes greater than one). This strategic assignment, where $I \text{ and } N \text{ are } < 1$ while $M > 1$, effectively expands the dynamic range, enhancing the overall contrast of the modified image.

3.1.1.4 Color correction

Color correction is a crucial post-processing technique that involves adjusting the colors in an image to achieve a more accurate and visually appealing representation. By manipulating the color balance, saturation, and brightness, color correction aims to eliminate any unwanted color casts and ensure that the colors in the image appear true to life. This process is commonly used in photography, graphic design, and video production to enhance the overall visual quality and consistency of the content. Sophisticated algorithms and software tools are often employed to precisely adjust individual color channels and achieve optimal color accuracy.

3.2 Leaf Localization

Leaf localization is a pivotal task in plant image analysis, aiming to identify and delineate the boundaries of leaves within images. Employed in agriculture, ecology, and plant biology, this process plays a crucial role in assessing plant health, disease detection, and growth monitoring. Computer vision algorithms, often utilizing convolutional neural networks (CNNs) and image processing techniques, are applied to accurately locate and segment leaves from complex backgrounds. By isolating leaf regions, researchers and farmers gain valuable insights into plant conditions. Leaf localization facilitates automated plant phenotyping, aiding in precision agriculture and advancing our understanding of plant responses to environmental factors.

3.2.1 Region Proposal Networks

The Region Proposal Network [29] streamlines object detection by generating rectangular proposals accompanied by objectness scores. Implemented as a fully-convolutional network, it shares convolutional layers with a Fast R-CNN detector. Two models, Zeiler and Fergus with 5 shareable convolutional layers, and Simonyan and Zisserman with 13, are explored. To generate proposals, a compact network traverses the final shared convolutional layer's output, linking to an n × n spatial window of the input feature map, as depicted in Fig 2. Each window corresponds to a lower-dimensional vector (256-d for ZF, 512-d for VGG), feeding into sibling fully-connected layers for box regression (reg) and box classification (cls). The architecture, with n = 3, employs shared fully-connected layers, realized through n × n conv followed by 1 × 1 conv layers for reg and cls. ReLUs enhance feature representation in the output of the n × n conv layer.

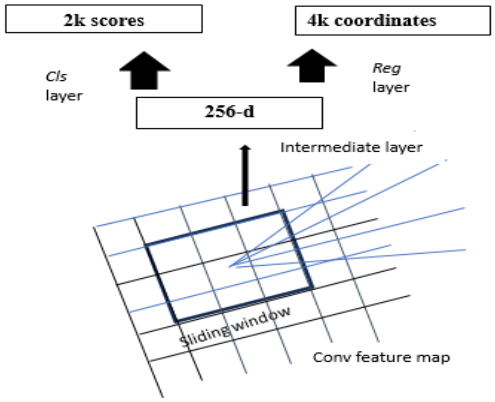


Fig 2: RPN [29]

3.2.2 Spatial Attention Mechanism (proposed)

Effectively identifying changed and unchanged areas relies on discriminant feature representations. Traditional Fully Convolutional Networks (FCNs) often generate local features, but relying solely on these may result in misclassification, as indicated by numerous studies. To address this limitation and capture the broader context of local features, we propose the integration of a spatial attention module [30]. This module plays a crucial role in encoding contextual information from long ranges into local features, thereby enhancing and enriching the overall feature representations. By incorporating spatial attention, our model gains the ability to better discern subtle changes and maintain improved accuracy in distinguishing between different types of features.

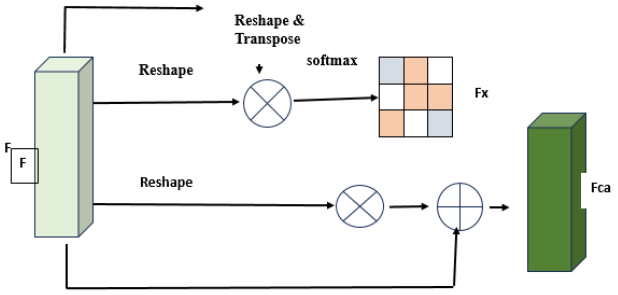


Fig 3: Spatial Attention Mechanism

As depicted in Fig. 3, the feature tensor $f \epsilon \mathbb{R}^{c \times h \times w}$ $with\ c$ representing the number of channels, $h \text{ and } w$ denoting the width and height, respectively, is derived from Siam-Conv. This feature is then fed into 3 convolutional layers sharing

the same structure, resulting in three new features: $f_a, f_b, and\ f_c$ where $\{f_a, f_b, and\ f_c\} \in \mathbb{R}^{c\times h\times w}$. Following this, $f_a, and\ f_b$ are reshaped to $\mathbb{R}^{c\times n}$, where $n = h \times w$.Subsequently, a matrix multiplication is performed between the transpose of $f_b\ and\ f_a$, , yielding the spatial attention map $f_s \in \mathbb{R}^{n\times n}$ through a softmax layer.

$$fs_{JI} = \frac{\exp(fa_I \cdot fb_J)}{\sum_{I=1}^{n} \exp(fa_I \cdot fb_J)} \tag{4}$$

fs_{JI} serves as a metric for evaluating the efficacy of the feature at position on the feature at position J. A higher value of fs_{JI} indicates a stronger connection between the two features.

After reshaping $fc\ to\ \mathbb{R}^{c\times n}$, perform matrix multiplication with $fs,$, yielding a result that is then reshaped to $\mathbb{R}^{c\times h\times w}$. Subsequently, the obtained result is multiplied by a scale parameter

η, followed by an elementwise summation operation with f, resulting in the final output. This process ensures the incorporation of spatial attention information for enhanced feature representation.

$$fsa_J = \eta \sum_{I=1}^{n} (fs_{JI} fc_J) + f_J \tag{5}$$

The parameter η initiates at 0 and dynamically adjusts to assign increasing weights. According to Formula (5), the resulting feature fsa at each position arises from a weighted sum that incorporates features from all positions and the original features. As a result, fsa captures a global context perspective, selectively aggregating contexts guided by spatial attention maps. This method ensures that similar semantic features reinforce one another, promoting compactness and semantic consistency within the class. Consequently, the network excels in distinguishing between genuine changes and pseudo-changes, thereby enhancing accuracy and discriminative capacity.

3.3. ROI Identification

The process of Region of Interest identification is facilitated through a comprehensive approach involving an Optimized Dual Attention YOLO model, The Feature Extraction is carried out using the FeatExProNet framework, encompassing various feature categories. Shape features, including metrics like area, major and minor axis length, perimeter, and solidity, provide insights into the geometric attributes of identified regions. Color features, employing histogram-based techniques, capture nuanced color distribution patterns within the regions of interest. Texture representation is enhanced through the utilization of Tamura and Haralick texture features. Statistical characteristics are encapsulated by extracting moments such as mean, skewness, and kurtosis. Furthermore, deep learning-based features are harnessed, leveraging Inception V3 for automatic and hierarchical extraction, ensuring a comprehensive understanding of regions of interest that combines geometric, color, texture, statistical, and deep learning-based attributes for robust ROI identification.

3.3.1 Optimized dual attention YOLO

In the optimization process of the Dual-Attention YOLO model, the focus lies on refining the model's parameters and improving convergence through strategic adjustments in the learning rate. Initially, the model architecture consists of a backbone and a head, with data augmentation performed by Maxup before backbone feature extraction. The backbone incorporates the GhostNet bottleneck and convolution structure, followed by outputs to the Vision Transformer block and the Spatial Pyramid Pooling Fusion layer. The GhostNet module efficiently reduces redundant information through linear operations, achieving model compression. Detailed structures are elucidated. The Vision Transformer block enhances the global receptive field on feature maps, capturing richer semantic information and provides a comprehensive understanding of its structure and operation. The SPPF layer, a spatial pyramid pooling layer, facilitates multi-scale information fusion by transforming input features into specific-dimensional vector information. The specific SPPF structure is detailed in Figure 6, incorporating convolution, concatenation, and max pooling operations. Deviating from the YOLOv5's FPN and PAN structure, the head network employs down-sampling through convolution output feature maps to enhance the receptive field. The Convolution-BatchNorm-ReLU structure is utilized, followed by the BiFPN feature fusion structure. This fusion mechanism, fully extracts input information of varying sizes for optimal fusion operations.

Augmenting multi-scale target recognition is attained through the CBAM channel spatial attention mechanism and the Vision Transformer block. The operational principle of the CBAM attention mechanism is elucidated. In the prediction phase, it is coupled with an enhanced K-Means algorithm for anchor clustering, contributing to improved accuracy and efficiency. Nine anchors of different sizes are obtained through clustering, with each group of three anchors of similar sizes classified together. Subsequently, predictions for large, medium, and small sizes are made, culminating in an improved K-Means anchor clustering process.

To optimize the Dual-Attention YOLO model with the learning rate, it is imperative to experiment with learning rate values, schedules, and other hyperparameters. Continuous monitoring of training progress and performance metrics enables iterative adjustments for enhanced convergence and model efficacy. The interplay between the intricate components of the model ensures a refined and well-tuned Dual-Attention YOLO architecture [34].

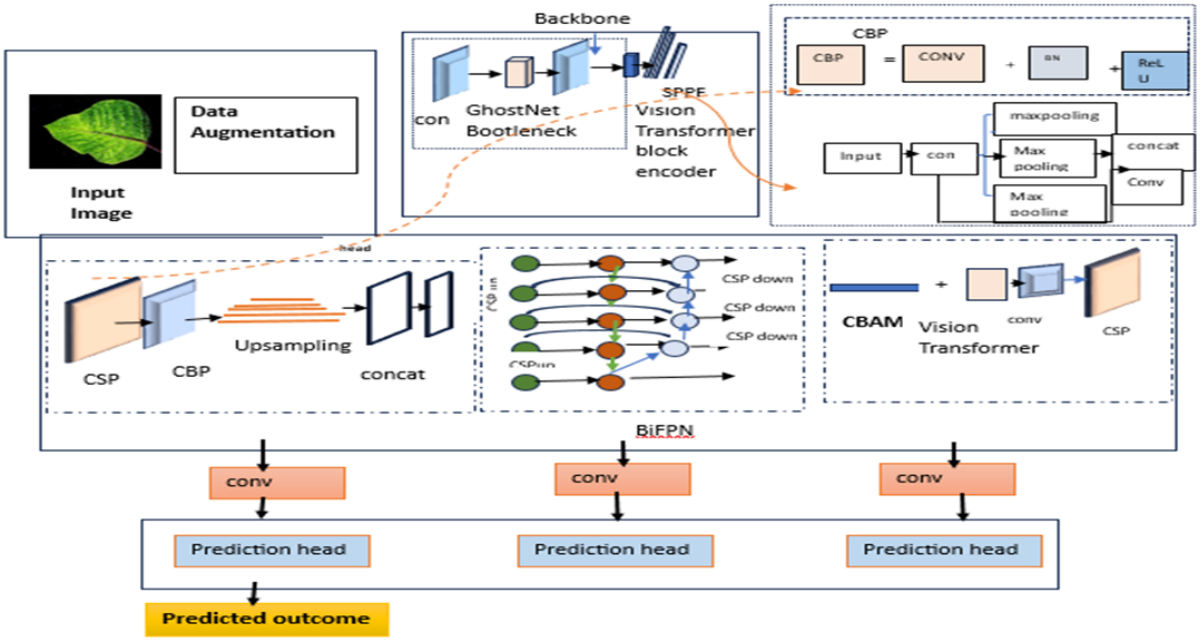


Fig 4: Optimized Dual Attention YOLO

3.3.2 FeatExProNet based Feature Extraction:

FeatExProNet employs a diverse Feature Extraction strategy encompassing various facets. Shape features, including area, major/minor axis length and perimeter offer geometric insights. Color analysis involves histogram-based features capturing distribution patterns. Texture representation is enhanced through Tamura and Haralick features. Statistical characteristics like mean, skewness, and kurtosis are extracted, and deep learning-based features leverage Inception V3 for automatic hierarchical extraction.

3.3.2.1 Shape

Extract features such as area, major/minor axis length, perimeter and solidity. Shape feature extraction is a crucial aspect of image analysis that involves capturing geometric characteristics to characterize objects or regions of interest. This process typically includes extracting parameters such as area, major/minor axis length, perimeter, and solidity. These features provide valuable information about the form, structure, and spatial layout of objects within an image, enabling effective discrimination and recognition in various applications, from computer vision to pattern recognition and object detection. The extracted shape features play a pivotal role in understanding and distinguishing different objects or regions based on their geometric attributes.

3.3.2.2 Area

Area quantifies the extent of a two-dimensional surface, representing the space enclosed by the boundary of a plane figure. It denotes the count of unit squares required to cover the closed figure's surface. Measurement units for area include square centimetres (cm^2) and square meters (m^2), offering a standardized means for expressing spatial coverage.

3.3.2.3 Perimeter

The perimeter of a shape is the complete distance around it, representing the total length if the shape were stretched linearly. It defines the boundary in a two-dimensional plane. Shapes with varying dimensions may share the same perimeter length, emphasizing how distinct shapes can exhibit equivalent perimeters based on their size and proportions.

$$perimeter = 2(L + B) \quad (6)$$

Where L is the length and B is the breadth.

3.3.2.4 Major and minor axis length

The major axis of an ellipse extends between two points on the curve, covering the maximum distance and accommodating both foci. In contrast, the minor axis connects the two co-vertices of the ellipse. If these co-vertices are positioned at (n,0) and (−n,0), the minor axis length equals 2n, emphasizing the geometric relationship within the ellipse.

3.3.2.5 Solidity

Solidity quantifies the relationship between the area of an object and the area of its convex hull. A solidity value of 1 signifies a solid object, while a value below 1 suggests irregular boundaries or the presence of voids. It's a metric comparing a polygon's area to the square of its perimeter, providing insights into the object's structural integrity.

3.3.2.6 Color

Utilize histogram-based features, capturing color distribution patterns. The extraction of histogram features initiates from the raw image data. These histograms undergo further processing to derive meta features, serving as input for the semantic mapper. The semantic mapper then transforms these meta features into semantic features. This methodology is outlined in the source publication titled "A Semantic Content-Based Retrieval Method for Histopathology Images."

3.3.2.7 Texture

Employ Tamura and Haralick texture features for enhanced texture representation. Tamura texture features are a set of measures designed to capture various aspects of texture in an image. Coarseness, one of the Tamura features, gauges the size of the texture elements. Contrast assesses the intensity variation between neighbouring pixels, while Directionality characterizes the predominant direction of texture patterns. These features collectively provide a detailed and discriminative description of the textural properties within an image, facilitating tasks such as image analysis, classification, and recognition.

Haralick texture features, derived from the gray-level co-occurrence matrix, offer a comprehensive characterization of image texture. These features include measures such as energy, quantifying the uniformity of pixel intensities; entropy, representing image randomness; contrast, indicating the intensity variation between pixels; and homogeneity, reflecting the closeness of pixel pairs in intensity. Haralick texture features are valuable for texture analysis in image processing, aiding tasks such as pattern recognition, segmentation, and classification by capturing intricate details within the textural composition of an image.

3.3.3 Statistical Features

Extract moments (mean, skewness, kurtosis) to capture statistical characteristics.

3.3.3.1 Mean

Calculate the mean by dividing the sum of numbers in a set by the total count of values, yielding the average. This statistical measure offers a central point of reference, encapsulating the collective magnitude of the dataset and providing a representative value for analysis.

$$\bar{Y} = \frac{\sum Y}{N} \tag{7}$$

To compute the arithmetic, mean of a dataset shown in Eq. (7), sum all data values (Y) using the symbol $\sum$ for summation. Divide the total by the number of values (N), where $\sum$ denotes summation. This method provides a representative average of the dataset.

3.3.3.2 Skewness

Skewness serves as a gauge of asymmetry within a distribution or dataset, specifically indicating its departure from symmetry. A distribution is considered symmetric when its visual representation mirrors on both sides of the central point. Skewness, therefore, quantifies the degree to which the data deviates from this balanced, symmetrical arrangement.

$$skew = \frac{\sum_{I=1}^{n}(y_I - \bar{y})^3/n}{S^3} \tag{8}$$

As per Eq. (8), Where mean value is defined as $\bar{y}$; the standard deviation is denoted as S; and the number of data point is referred as n.

3.3.3.3 Kurtosis

Kurtosis functions as an indicator of the tail behavior of data concerning a normal distribution. It discerns whether the data exhibit heavier or lighter tails compared to a standard normal distribution. This metric provides valuable insights into the shape and characteristics of the distribution, offering information about the data's propensity for extreme values or outliers.

$$kurtosis = \frac{\sum_{I=1}^{n}(y_I - \bar{y})^4/n}{S^4} \tag{9}$$

As per Eq. (9), Where mean value is defined as $\bar{y}$; the standard deviation is denoted as S; and the number of data point is referred as n.

3.3.4 Deep Learning-based Features

Leverage Inception V3 for automatic and hierarchical feature extraction.

3.3.4.1 Inception V3

Incorporating over 20 million parameters, the inception-V3 model boasts a robust architecture crafted by a prominent hardware expert. Comprising symmetrical and asymmetrical blocks, it features diverse layers—convolutional, average and max pooling, concatenations, dropouts, and fully connected layers. The consistent application of batch normalization enhances activation layer inputs. The classification process utilizes Softmax, ensuring efficient categorization. The model's complexity and effectiveness are underscored by the intricate interplay of these elements, contributing to its superior performance. Figure 5 provides a schematic diagram, offering a visual representation of the Inception-V3 model's architectural intricacies, showcasing its capacity to capture intricate features and patterns in diverse datasets.

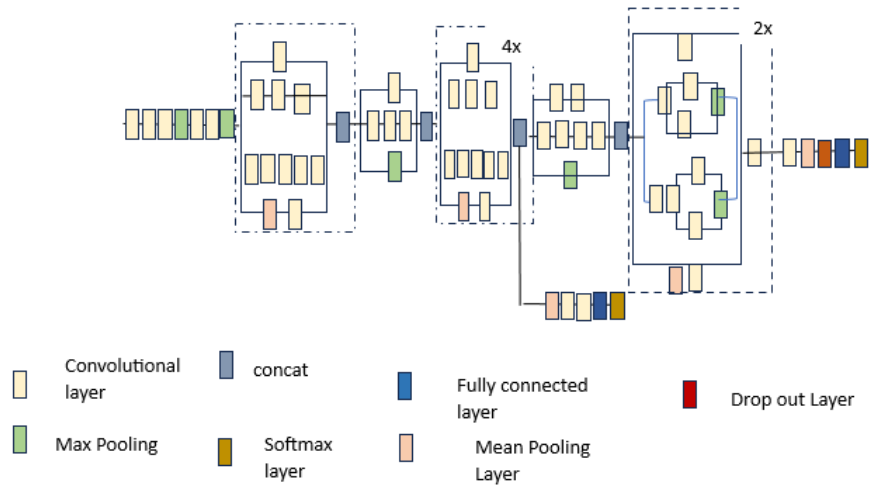


Fig 5: Inception V3 architecture

3.4 Feature Selection

In the realm of feature selection, a novel strategy unfolds through a Hybrid Optimization Approach. This innovative method seamlessly integrates Binary Sand Cat Swarm Optimization and Butterfly Optimization algorithms. By synergizing these techniques, the approach enhances the efficiency of feature selection, ensuring a refined and effective feature subset for optimal model performance.

3.4.1 Binary Sand Cat Swarm Optimization

In the Sand Cat Search Optimization (SCSO) algorithm, the population comprises N sand cat individuals with D dimensions, forming an N × D matrix. The matrix, denoted as X(t), encapsulates the position vectors of sand cats in the search space at iteration t, representing the evolving solutions throughout the optimization process.

The Sand Cat in SCSO possesses a sensitivity range (rg) for low-frequency noises from 2 kHz to 0 kHz. This sensitivity decreases linearly as the frequency diminishes. The calculation of rg captures the dynamic nature of the Sand Cat's auditory sensitivity, reflecting its responsiveness to varying frequency levels during the optimization process.

$$RG = S_m - \left(\frac{S_m \times t}{T}\right) \quad (10)$$

SettingS_m as 2, the current iteration (t) and maximum iterations (T) influence the R parameter, determining the exploration-exploitation trade-off in SCSO.

$$r = \left((2 \times RG) \times rand(0,1)\right) - RG \quad (11)$$

Utilizing $rand(0,1)$ to generate a random number, the R parameter in SCSO, defining sensitivity range for potential solutions, is computed accordingly.

$$R = RG \times rand(0,1) \quad (12)$$

The Sand Cat Search Optimization (SCSO) determines the next location based on R (-1 to 1). When $|r| \leq 1$, the approach prioritizes exploitation for precise prey hunting. Conversely, if |R| > 1, the algorithm emphasizes exploration, compelling sand cats to seek new food sources. The mathematical expression for prey attack (exploitation) in SCSO is detailed accordingly.

$$x_{rand} = |rand(0,1) \times x_{best} - x(T)| \quad (13)$$

$$x_{(t+1)} = x_{best} - rand(0,1) * x_{rand} * \cos(\theta) \quad (14)$$

In this equation (13-14), x_{rand} computes the distance from the best position, Xbest, to the current position, $x(T)$, in iteration t. The resulting $x_{(t+1)}$signifies the updated position of the search agent, indicating the movement of the sand cat. Additionally, the circular sensitivity of sand cats directs movement through a random angle θ determined by roulette wheel selection Eq. (15-17).

The exploration phase in SCSO is expressed as the mathematical formulation for searching prey, guiding the sand cat in exploration endeavours.

$$CP = floor(n * rand(0,1) + 1) \quad (15)$$

$$x_{candidate}(t) = x(cp,:) \quad (16)$$

$$x_{(t+1)} = R \times (x_{candidate}(t) - rand(0,1) \times x(t)) \quad (17)$$

$x_{candidate}(t)$ represents a randomly chosen candidate position in the exploration phase.

In the realm of feature selection, envisioning each feature as a binary decision—either included or excluded in the final subset—translates into a binary vector of size D, where D signifies the total features. A value of 1 denotes inclusion,

while 0 signifies exclusion. SCSO operates in a continuous space, but the feature selection problem operates in a discrete space. To bridge this gap, transfer functions are employed to transform the continuous space into a discrete one, enabling the adaptation of the SCSO algorithm for feature selection.

3.4.2 Butterfly Optimization

The Butterfly Optimization Algorithm is a swarm-based metaheuristic that draws inspiration from the foraging and mating behaviour of butterflies. This innovative algorithm is grounded in three fundamental hypotheses, each contributing to the optimization process.

Firstly, BOA posits that all butterflies emit fragrance and are naturally drawn to one another. This collective attraction forms the basis for the swarm's cohesion during optimization. Secondly, individual butterflies are assumed to move randomly or towards the butterfly emitting the most scent, mirroring the exploration-exploitation trade-off in the search space. This stochastic movement ensures a diverse exploration of the solution landscape. Lastly, BOA suggests that the stimulus intensity experienced by a butterfly is intricately linked to the fitness landscape. As butterflies traverse the search space, the changing fragrance levels act as indicators of the optimization landscape.

The optimization process in BOA unfolds through two distinct phases: the local search phase and the global search phase. During the global search, butterflies navigate randomly when they do not detect the fragrance network. This phase facilitates broad exploration of the solution space. In contrast, the local search phase kicks in as butterflies converge towards the individual emitting the highest concentration of fragrance. This targeted movement intensifies exploration around promising regions, refining the search for optimal solutions.

Mathematically, BOA employs a model that captures the dynamics of fragrance emission, movement, and stimulus intensity. This model facilitates the integration of global and local search strategies, enabling the algorithm to effectively address optimization problems. In essence, the Butterfly Optimization Algorithm stands out as a unique approach, leveraging the collective behavior of butterflies to inspire a powerful optimization technique that adeptly balances exploration and exploitation in search spaces.

Butterfly fragrance correlates with the stimulus intensity, forming a function that reflects the dynamic interplay of their environmental interactions.

$$F_I = Ci^a, I = 1,2, \ldots . np \quad (18)$$

In the mathematical model, butterfly fragrance (F) is a function of sensory modality (C), stimulus intensity (I), power exponent (a), and the number of butterflies (np). The BOA model encapsulates the dynamic interplay of these factors during both global and local search phases, offering a comprehensive representation of how butterflies collectively explore and exploit the optimization landscape.

$$x_I^{T+1} = x_I^T + (R^2 \times x_{best}^T - x_I^T) \times F_I \quad (19)$$

$$x_I^{T+1} = x_I^T + (R^2 \times x_J^T - x_K^T) \times F_I \quad (20)$$

In the algorithm, the position of the $I-th$ butterfly during the $T-th$ iteration is denoted by x_I^T. The global optimal individual is represented by x_{best}^T. Random number $R \in (0,1)$, alongside randomly selected individuals x_J^T and x_K^T, influences the search. BOA employs two search strategies, controlled by a switching probability $\boldsymbol{P}$, managing the dynamic transition between these approaches during the optimization process.

3.5 VarioFusionNet-based leaf disease detection:

In the realm of feature selection, a novel strategy unfolds through a Hybrid Optimization Approach. This innovative method seamlessly integrates Binary Sand Cat Swarm Optimization and Butterfly Optimization algorithms. By synergizing these techniques, the approach enhances the efficiency of feature selection, ensuring a refined and effective feature subset for optimal model performance.

3.5.1 Vision Transformer

The Vision Transformer exhibits advantages, avoiding saturation with increasing model depth and dataset size. Its capacity to process sequences of any length within memory constraints stands out. While convolutional neural networks like ResNet excel in small to medium image tasks, Vision Transformer introduces global self-attention on feature graphs, lacking some inductive bias present in CNNs. As data volume grows, Transformer's strengths become more apparent. In this study, Vision Transformer replaces segments of YOLOv5's network structure, demonstrating superior recognition performance, showcasing the potential synergy between Transformer architectures and object detection tasks.

The Transformer-based pure encoder structure involves embedding a 1D vector in the sequence input, addressing challenges in capturing global characteristics. Preserving image position information is achieved through 1D position embedding, learnable by a linear layer n.

However, executing global self-attention in entity objects leads to a significant $O(n\text{^}2d)$ computational complexity for Transformer. Vision Transformer's encoder comprises two independent sub-layers: the MLP layer and the multi-head self-attention network. Each floor l performs operations on input $i^{L-1} \epsilon\, R^{l \times c}$, , generating (q, k, v) triples for subsequent self-attention processing. This architecture enhances the network's ability to capture intricate relationships in data.

$$q = i^{L-1}, k = i^{L-1}w_K, v = i^{L-1}w_V \quad (21)$$

In the self-attention (SA) process, denoted by Figure 7b's yellow section, three weight vectors w_q, w_K, w_V correspond to linear mappings, with d representing the feature vector's dimension. This process involves mapping input features to query (q), key (k), and value (v) vectors, essential for capturing intricate relationships within the data through attention mechanisms.

$$sa(i^{L-1}) + softmax(\frac{i^{L-1}w_q(zw_K)^t}{\sqrt{D}})(z^{L-1}w_V) \quad (22)$$

A multi-head mechanism is formed by connecting n self-attention modules in series. The output, denoted as $MSA(\ i^{L-1})$, is a concatenation of the individual self-attention module outputs. This concatenated output is transformed by a weight matrix $w_o \in R^{md} \times c$, where d is c/m. The resulting output, along with a residual connection, serves as input for the MLP layer.

3.5.2 GoogLeNet[32]

The GoogLeNet network model aims to enhance network width through its prominent Inception structure, as depicted in Figure 1. This structure strategically employs 1x1 convolutional kernels for dimensionality reduction, effectively reducing parameters and increasing network depth. The branching and merging architecture in Figure 6 contribute to expanding the network width, thereby improving accuracy. Inception-v1, outlined in Table 1, delineates the network structure, detailing types, depth, pooling, fully connected layers (fc), and the softmax output layer for probability results.

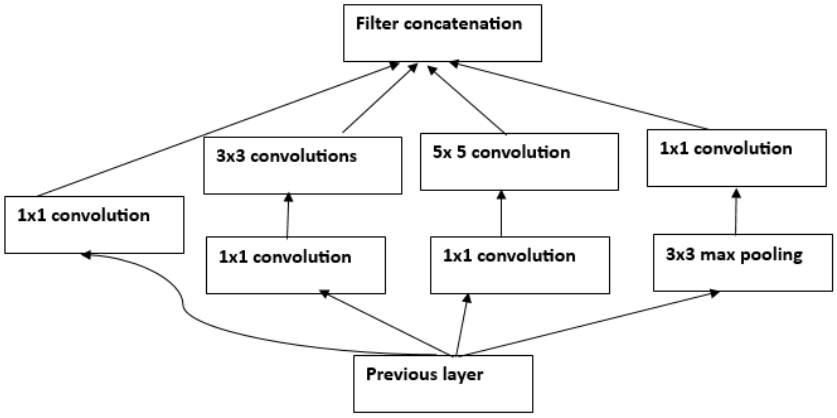


Fig 6: GoogLeNet Architecture

Following Inception-v1, the GoogLeNet network has seen continuous refinement, leading to subsequent versions. Inception-v2 introduces batch normalization, Inception-v3 substitutes two-dimensional convolution kernels with one-dimensional counterparts, and Inception-v4 incorporates ideas from the residual network concept. This article adopts the Inception-v4 structure for its advanced features. The evolution of GoogLeNet signifies a commitment to refining architectures for optimal accuracy and performance in image recognition tasks, demonstrating the continuous advancements in deep learning network design and the ongoing quest for more efficient and effective models in the field.

3.5.3 Alex Net

AlexNet, detailed in [33], consists of five convolutional layers. The first four layers are succeeded by pooling layers, while the fifth integrates three fully-connected layers. Back-propagation, using stochastic gradient descent, fine-tunes convolutional kernels to minimize the overall cost function. Sliding kernels in the convolutional layers operate on input feature maps, producing convolved feature maps. Additionally, pooling layers perform max or average pooling operations to aggregate information within neighbourhood windows.

Alex Net's triumph is credited to key strategies like the ReLU non-linearity layer and dropout regularization. The ReLU function, expressed in Equation (23), accelerates training and prevents overfitting by acting as a half-wave rectifier. Dropout, prominently used in fully connected layers, acts as regularization by randomly zeroing some input or hidden neurons, alleviating co-adaptations. These combined tactics enhance Alex Net's efficacy in image classification tasks.

$$f(X) = \max(X, 0) \quad (23)$$

The efficacy of transferring CNN parameters from natural imagery to HSR remote sensing datasets hinges on dataset similarities and category compatibility. Utilizing well-trained parameters from complex datasets like ImageNet, the pre-training mechanism ensures a robust start for AlexNet. This initialization is pivotal for subsequent HSR remote sensing scene classification, showcasing the

architecture's versatility. The pre-training transforms AlexNet into an end-to-end classification pipeline, streamlining HSR remote sensing imagery processing. Its adaptability underscores the model's efficiency in handling diverse data, affirming its role as a powerful tool for intricate tasks in high-resolution remote sensing applications.

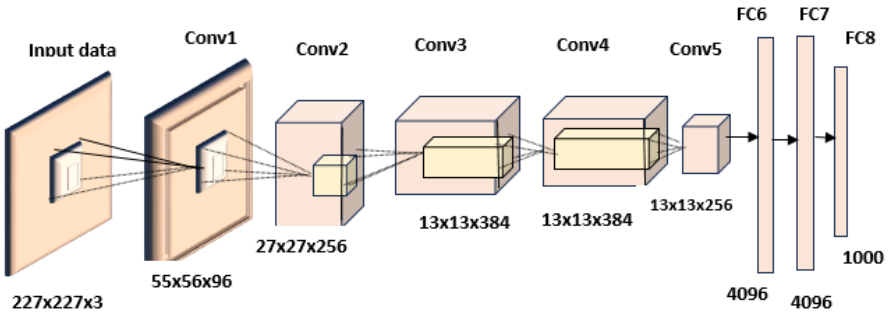


Fig 7: AlexNet Architecture

3.5.4 DenseNet 121

Within DenseNet architectures, layers form dense blocks, ensuring each layer incorporates inputs from all preceding layers. The $L-th$ layer receives input from all previous feature maps $[X_0, X_1, \ldots\ldots, X_{L-1}]$, facilitating the creation of a comprehensive feature map that is then propagated to subsequent layers. This interconnected design enhances information flow and promotes feature reuse.

$$X_L = h_L([X_0, X_1, \ldots\ldots, X_{L-1}]) \tag{24}$$

In the DenseNet architecture, $[X_0, X_1, \ldots\ldots, X_{L-1}]$ consolidates previous feature maps for the l-th layer. X_L, the output of this layer, is generated by the composition function h_L, involving batch normalization, ReLU activation, and convolution. Differentiating from methods like ResNet, DenseNet concatenates layers rather than combining past and future layers. This approach combats vanishing gradients by reusing features, effectively reducing parameters. DenseNet-121 features four dense blocks interspersed with transition layers employing down-sampling through 1×1 convolution and 2×2 average pooling. Cross-layer connections in dense blocks enhance non-linearity through the utilization of ReLU activation.

The ReLU activation proposed for increased non-linearity is defined succinctly.

$$ReLU(X) = Max(0, X) \tag{25}$$

The ultimate layer employs a fully connected layer with a softmax function for weather image class probability prediction, defined as follows:

$$Sm(Z)_I = \frac{e^{Z_I}}{\sum_{J=1}^{C} e^{Z_J}} \; for \; I = 1,2 \ldots . C \tag{26}$$

In this context, the input vector $Z = (Z_1, \ldots\ldots\ldots, Z_C) \in \mathbb{R}^C$, undergoes exponentiation for each zi, ensuring the output vector $Sm(Z)$ sums to 1.

3.5.5 ResNet 50

Earning the ImageNet Large Scale Visual Recognition Challenge's top spot, the Residual Neural Network reshaped deep learning. It introduced a groundbreaking approach by incorporating residual connections between layers. This innovation involves the output of a layer being a convolution of its input and the input itself. This distinctive design mitigates loss, preserves knowledge gains, and significantly improves overall training performance. The Residual Neural Network's success stems from its ability to address the challenges of training deep networks, marking a paradigm shift in neural network architectures and establishing itself as a cornerstone in the evolution of deep learning. This unique architectural feature empowers ResNet to effectively tackle challenges in deep learning tasks. Fig 8., showcases a block diagram elucidating the distinctive architecture of the ResNet model [31], underlining its groundbreaking design in achieving superior performance in image recognition.

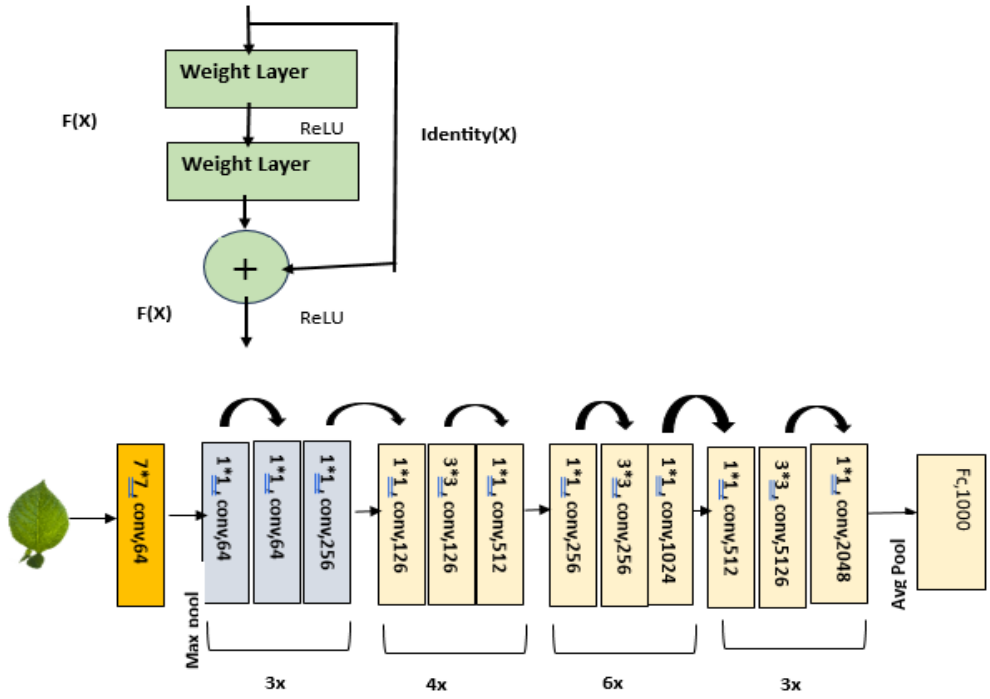


Fig 8: ResNet 50 Architecture

3.5.6 EfficientNet

Incorporating Mobile Inverted Bottleneck layers, EfficientNet combines depth-wise separable convolutions and inverted residual blocks for optimal efficiency. Performance is further heightened by integrating Squeeze-and-Excitation optimization. The MBConv layer, drawing inspiration from MobileNetV2's inverted residual blocks, initiates with a depth-wise convolution. It is succeeded by a point-wise convolution for channel expansion and culminates with another 1x1 convolution for channel reduction. This fusion of techniques within EfficientNet underscores its sophisticated architecture, resulting in improved computational efficiency and robust model performance. This design balances efficiency and representational power. EfficientNet further incorporates SE blocks, leveraging global average pooling and fully connected layers to refine feature maps, emphasizing crucial information. The model offers variants like EfficientNet-B0, B1, etc., each presenting a tailored balance between model size and accuracy for diverse user needs.

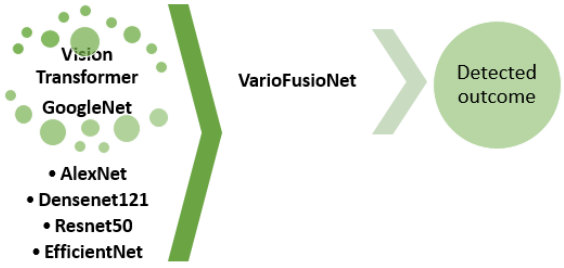


Fig 9: VarioFusionNet architecture

4. Result and Discussion

In this research the following section elaborates the result and discussion of various leaf disease detection.

4.1 Experimental Setup

Implemented in the experimental setup, the proposed model undergoes evaluation Accurate and Timely Leaf Disease Detection and Management tool in MATLAB. Key performance metrics, including sensitivity, specificity, accuracy, precision, FPR, FNR, NPV, F-Measure, MCC, and Recall, are meticulously considered or the wheat, pepper and tomato leaf providing a thorough assessment of the model's efficacy and the images of leaf localization, pre-processed and ROI are shown in Fig. 13-15.

The performance analysis of the proposed VarioFusionNet model for wheat classification is presented in Table 2, alongside comparisons with other established models, including Google Net, ResNet50, Efficient Net, LSTM, and GRU. The proposed VarioFusionNet exhibits superior sensitivity at 0.9878, indicating a high ability to correctly identify wheat instances. Specificity is notably high at 0.9939, demonstrating the model's ability to identify non-wheat objects. The overall accuracy stands at 0.991, emphasizing the model's precision across both wheat and non-wheat classes. In terms of precision, the proposed VarioFusionNet achieves a commendable 0.987, highlighting its accuracy in correctly predicting wheat instances. The recall rate of 0.987 reflects the model's capability to capture a substantial portion of actual wheat instances. The recall, harmonizing precision and, F-Measure is at 0.987, affirming a balanced performance. Negative Predictive Value (NPV) is high at 0.9939, indicating the

model's ability in accurately identifying non-wheat instances. The False Positive Rate (FPR) is impressively low at 0.0061, signifying minimal instances where non-wheat is misclassified as wheat. The False Negative Rate (FNR) is equally low at 0.0122, demonstrating rare cases of wheat being overlooked. The Matthews Correlation Coefficient (MCC) of 0.9817 attests to the overall strength of the proposed VarioFusionNet in wheat classification. Comparatively, GoogLeNet, ResNet50, EfficientNet, LSTM, and GRU exhibit respectable performances, yet the proposed VarioFusionNet emerges as a promising model, outperforming its counterparts across various key metrics.

Table 2: Performance analysis of the proposed model for wheat

Wheat	Sensitivity	Specificity	Accuracy	Precision	Recall	FMeasure	NPV	FPR	FNR	MCC
Proposed VarioFusionNet	0.9878	0.9939	0.991	0.987	0.987	0.987	0.9939	0.0061	0.0122	0.9817
GoogleNet	0.9755	0.9878	0.983	0.975	0.9755	0.975	0.9878	0.0122	0.0245	0.9633
ResNet50	0.9688	0.9844	0.979	0.968	0.9688	0.968	0.9844	0.0156	0.0313	0.9531
EfficientNet	0.9579	0.9789	0.971	0.957	0.9579	0.957	0.9789	0.0211	0.0421	0.9368
LSTM	0.9538	0.9769	0.969	0.953	0.9538	0.953	0.9769	0.0231	0.0462	0.9307
GRU	0.9497	0.9749	0.966	0.949	0.9497	0.949	0.9749	0.0251	0.0503	0.9246

Table 3 presents a comprehensive performance analysis of the proposed VarioFusionNet model for the classification of peppers, comparing its efficacy with other well-established models such as GoogLeNet, ResNet50, EfficientNet, LSTM, and GRU. The proposed VarioFusionNet demonstrates a notable sensitivity of 0.9832, indicative of its ability to effectively identify true positive instances of peppers. Its specificity is also high at 0.9848, showcasing proficiency in discerning non-pepper entities. The overall accuracy of 0.983 underscores the model's precision in classifying both pepper and non-pepper instances. The precision of the proposed VarioFusionNet is commendable at 0.989, highlighting its accuracy in correctly predicting instances of peppers. The recall rate, standing at 0.9832, signifies the model's capability to capture a significant proportion of actual pepper instances. The F-measure, a harmonic mean of precision and recall, is robust at 0.9865, attesting to the model's balanced performance. Negative Predictive Value (NPV) is high at 0.9749, indicating the model's efficacy in accurately identifying non-pepper instances. The False Positive Rate (FPR) is impressively low at 0.0152, suggesting minimal instances where non-pepper entities are misclassified as peppers. The False Negative Rate (FNR) is similarly low at 0.0168, indicating rare cases where peppers are overlooked. The Matthews Correlation Coefficient (MCC) for the proposed VarioFusionNet is noteworthy at 0.9664, affirming the model's general resilience in the pepper classification. Comparative analysis with GoogleNet, ResNet50, EfficientNet, LSTM, and GRU reveals that the proposed VarioFusionNet consistently outperforms its counterparts across various crucial metrics, emphasizing its promise in pepper classification.

Table 3: Performance analysis of the proposed model- Peppers

Peppers	Sensitivity	Specificity	Accuracy	Precision	Recall	FMeasure	NPV	FPR	FNR	MCC
Proposed VarioFusionNet	0.9832	0.9848	0.983	0.989	0.9832	0.9865	0.9749	0.0152	0.0168	0.9664
GoogleNet	0.9798	0.9747	0.977	0.983	0.9798	0.9815	0.9698	0.0253	0.0202	0.9538
ResNet50	0.9731	0.9646	0.969	0.976	0.9731	0.9747	0.9598	0.0354	0.0269	0.9369
EfficientNet	0.9697	0.9596	0.965	0.973	0.9697	0.9713	0.9548	0.0404	0.0303	0.9285

LSTM	0.963	0.9495	0.957	0.966	0.963	0.9646	0.9447	0.0505	0.037	0.9117
GRU	0.953	0.9391	0.947	0.959	0.953	0.9562	0.9296	0.0609	0.047	0.8906

In the evaluation of the proposed VarioFusionNet model for tomato classification, Table 4 provides a comprehensive performance analysis alongside comparisons with established models such as Google Net, ResNet50, Efficient Net, LSTM, and GRU. The proposed VarioFusionNet demonstrates high sensitivity at 0.9522, indicating its effectiveness in identifying true positive instances of tomatoes. Specificity is notably strong at 0.9947, showcasing the model's ability to accurately distinguish non-tomato entities. The overall accuracy is impressive at 0.9904, underscoring the model's precision in classifying both tomato and non-tomato instances. In terms of precision, the proposed VarioFusionNet achieves a commendable 0.9522, highlighting its accuracy in correctly predicting instances of tomatoes.

With a recall rate of 0.9522, the model adeptly captures a substantial portion of actual tomato instances. The F-measure, harmonizing precision and recall, remains robust at 0.9522, attesting to the model's balanced performance. Notably high Negative Predictive Value (NPV) at 0.9947 showcases the model's proficiency in accurately identifying non-tomato instances. Impressively low False Positive Rate (FPR) at 0.0053 indicates rare misclassifications of non-tomato entities as tomatoes. Equally low False Negative Rate (FNR) at 0.0478 suggests infrequent instances of overlooked tomatoes. The Matthews Correlation Coefficient (MCC) for VarioFusionNet is noteworthy at 0.9469, confirming the model's overall robustness in tomato classification. Comparative analysis with Google Net, ResNet50, Efficient Net, LSTM, and GRU reveals that the proposed VarioFusionNet consistently outperforms its counterparts across various crucial metrics, emphasizing its promise in tomato classification.

Table 4: Performance analysis of the proposed model-tomato

Tomato	**Sensitivity**	**Specificity**	**Accuracy**	**Precision**	**Recall**	**FMeasure**	**NPV**	**FPR**	**FNR**	**MCC**
Proposed VarioFusionNet	0.9522	0.9947	0.9904	0.9522	0.9522	0.9522	0.9947	0.0053	0.0478	0.9469
GoogleNet	0.9132	0.9904	0.9826	0.9132	0.9132	0.9132	0.9904	0.0096	0.0868	0.9035
ResNet50	0.8682	0.9854	0.9736	0.8682	0.8682	0.8682	0.9854	0.0146	0.1318	0.8536
EfficientNet	0.8567	0.9841	0.9713	0.8567	0.8567	0.8567	0.9841	0.0159	0.1433	0.8407
LSTM	0.7983	0.9776	0.9597	0.7983	0.7983	0.7983	0.9776	0.0224	0.2017	0.7758
GRU	0.7811	0.9757	0.9562	0.7811	0.7811	0.7811	0.9757	0.0243	0.2189	0.7567

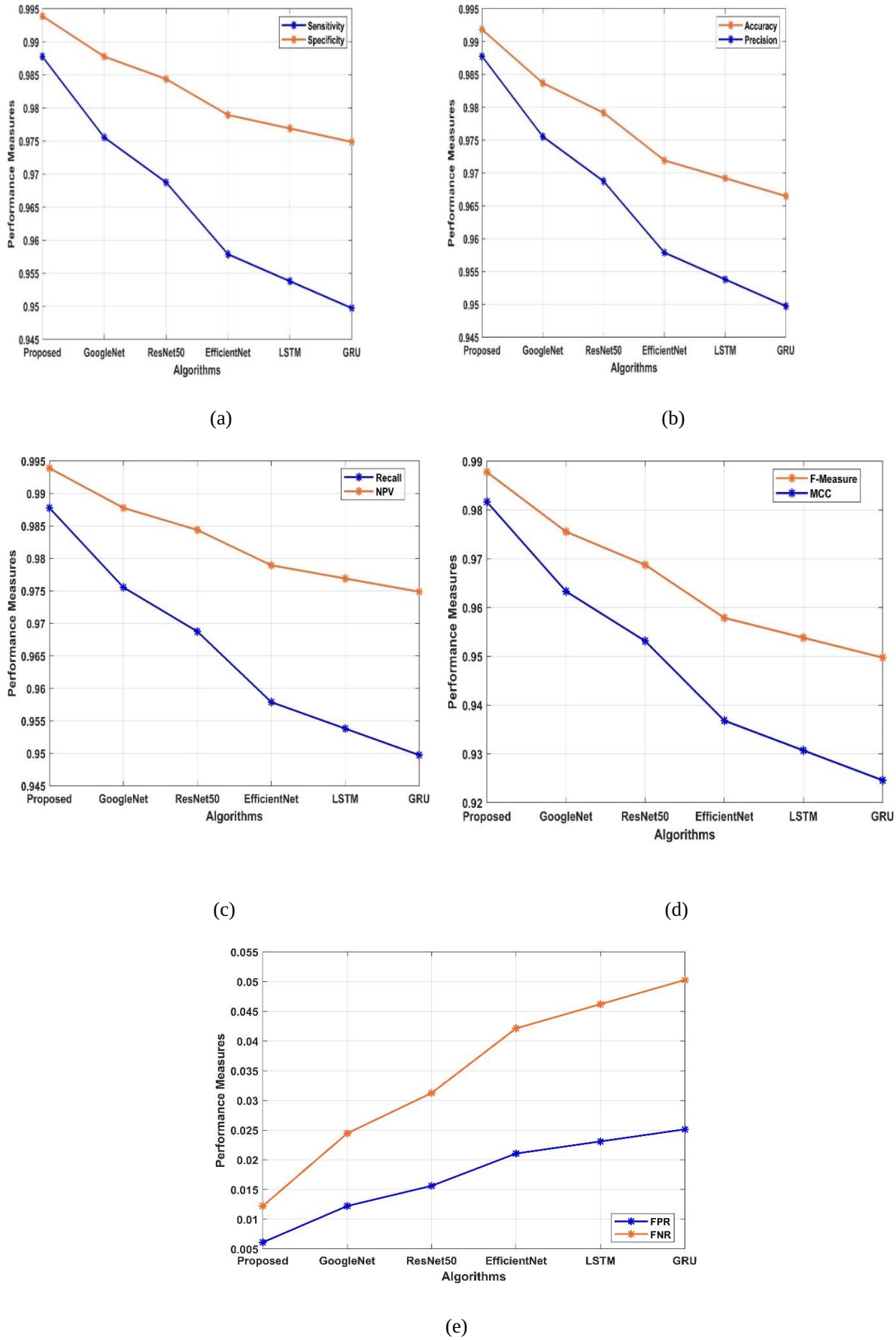
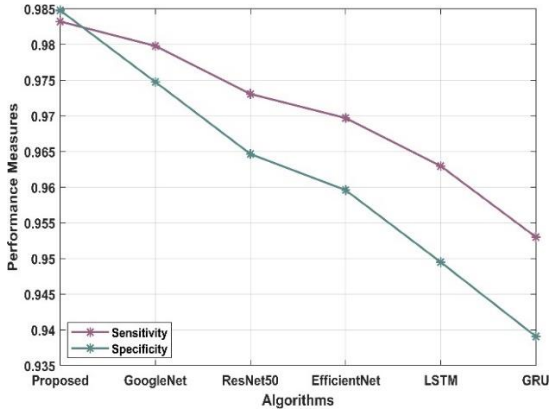
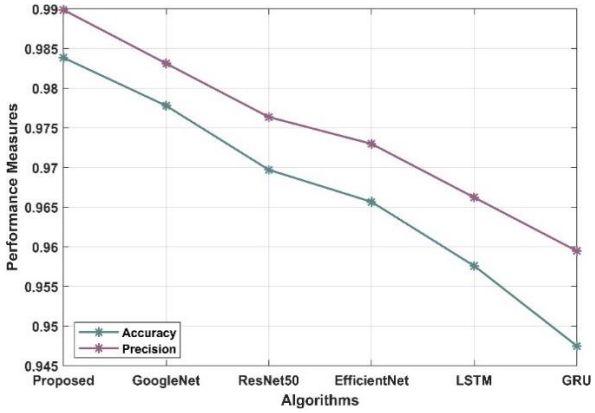


Fig 10: (a)- (e) Pictorial format of the performance metrics of the wheat leaf basis of the suggested model

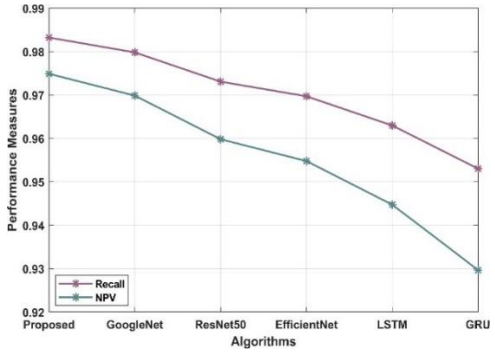
Figure 10 (a) –(e) illustrates the graphical format of the performance metrics of the proposed with other comparable models in the observation of wheat leaf. It is clear that the proposed model recall, sensitivity, precision, specificity, accuracy, MCC, F-measure, and NPV value are high and at the same time the FPR and FNR value are low when compared to the other comparable models like Google Net, ResNet50, Efficient Net, LSTM and GRU, these are due to the apply of hybrid optimization model.



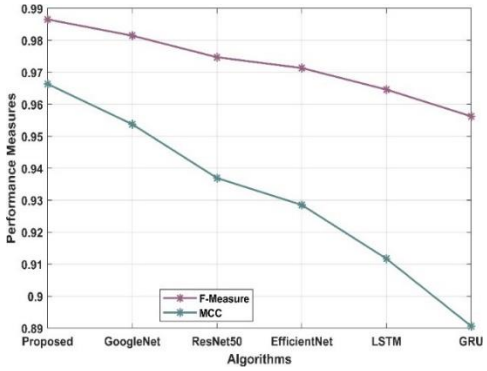
(a)



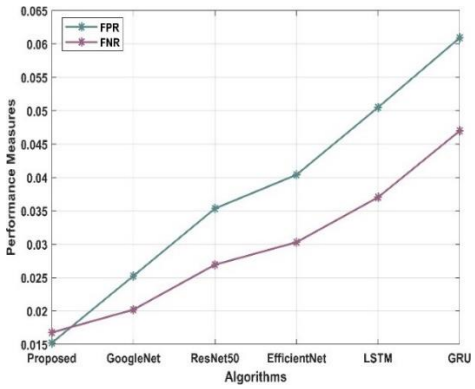
(b)



(c)



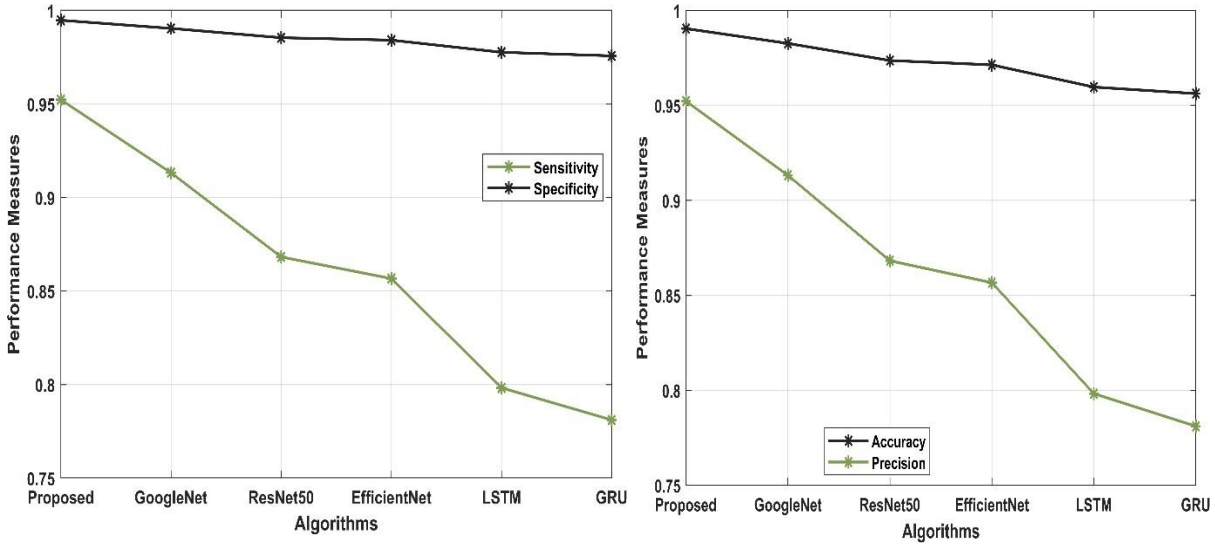
(d)



(e)

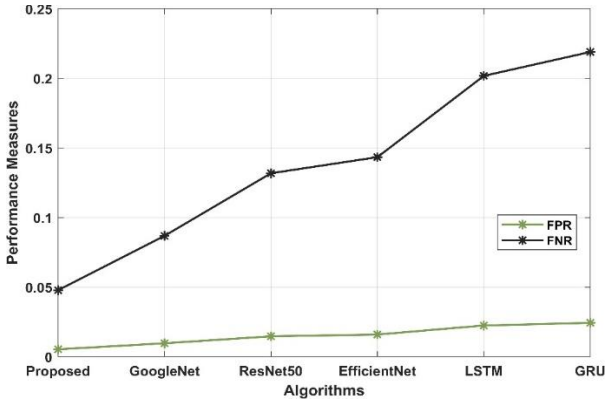
Fig 11: (a)- (e) Pictorial format of the performance metrics of the pepper leaf basis of the proposed model

In Figures 11 (a)–(e), the graphical representation of performance metrics for wheat leaf observation illustrates the superiority of the proposed model over comparable models like Google Net, ResNet50, Efficient Net, LSTM, and GRU. The proposed model exhibits high sensitivity, specificity, accuracy, precision, MCC, F-measure, recall, and NPV values, coupled with low False Positive Rate and False Negative Rate. This superior performance is attributed to the application of a hybrid optimization model.



(a) (b)

(c) (d)



(e)

Fig 12: (a)- (e) Pictorial format of the performance metrics of the tomato leaf basis of the proposed model

In the depicted Figures 12 (a)–(e), the visual representation of performance metrics in wheat leaf observation underscores the superior performance of the proposed model compared to counterparts like Google Net, ResNet50, Efficient Net, LSTM, and GRU. The proposed model showcases elevated sensitivity, specificity, accuracy, precision, MCC, F-measure, recall, and NPV values, complemented by minimal False Positive Rate and False Negative Rate. This exceptional performance is credited to the integration of a hybrid optimization model.

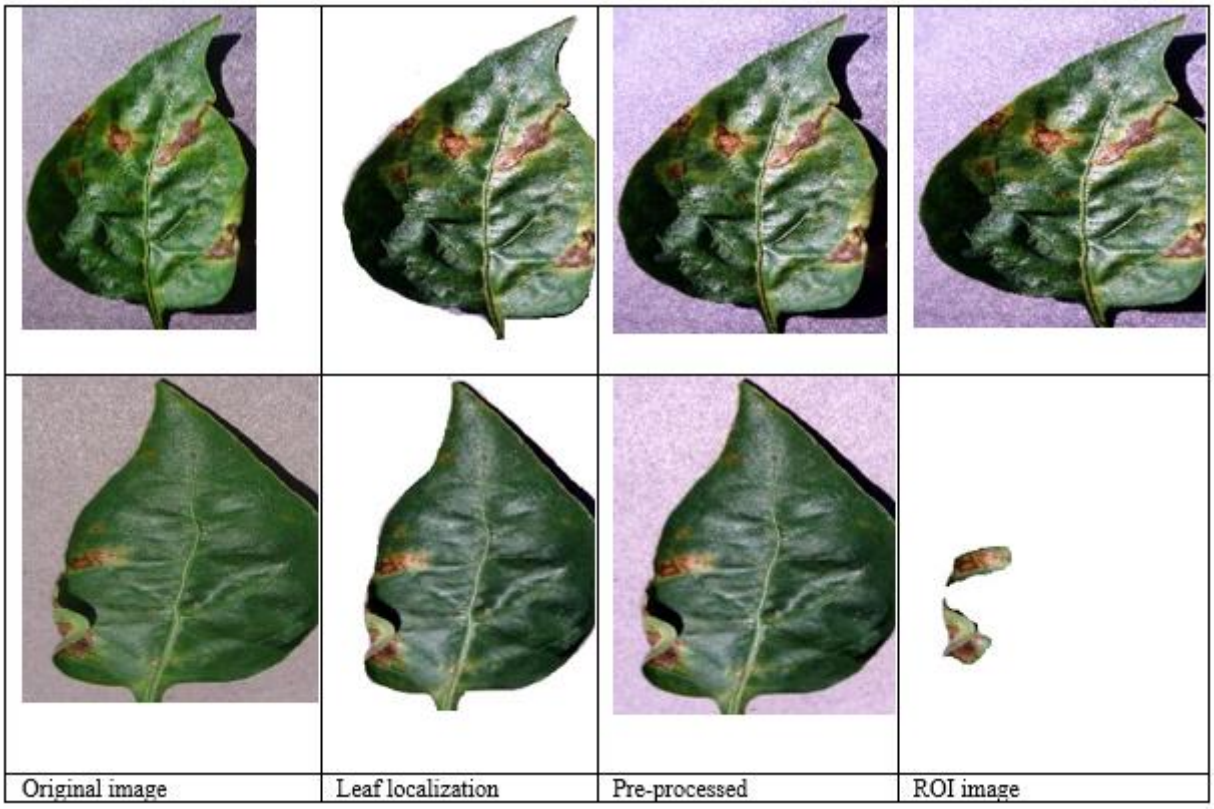


Fig 13: Pepper leaf localization, pre-processed and ROI image representation

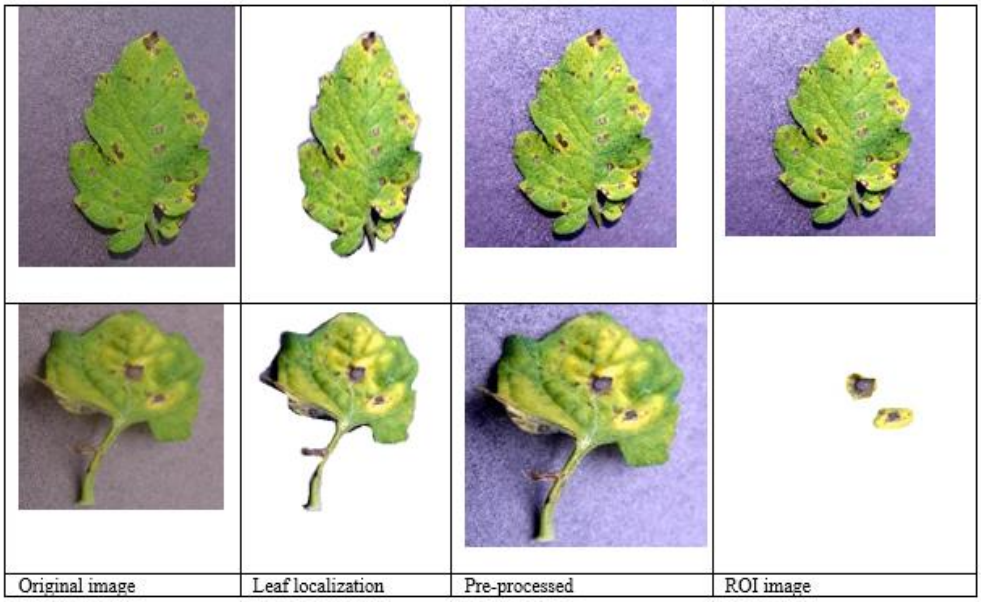


Fig 14: Tomato leaf localization, pre-processed and ROI image representation

Fig 15: Wheat leaf localization, pre-processed and ROI image representation

5. Conclusion

The methodology unfolds with a strategic execution of the outlined objectives, commencing with the application of advanced pre-processing techniques and concluding with the validation on diverse datasets. Each stage is meticulously designed, prioritizing the incorporation of sophisticated deep learning methodologies to ensure accuracy throughout the process. In the initial step of pre-processing, the dataset is fortified through the utilization of Generative Adversarial Networks for image augmentation, elevating the model's robustness. A novel Modified Gaussian Smoothing technique is then applied for noise reduction and enhanced image quality, while Contrast Stretching is employed to optimize image contrast, facilitating superior feature extraction. Additionally, color correction methods are implemented to standardize color variations across the dataset. Moving to the leaf localization phase, the integration of Region Proposal Networks with proposed Spatial Attention Mechanisms significantly enhances the accuracy of leaf localization, ensuring precise identification of disease-affected areas. Subsequently, the Region of Interest identification step incorporates an optimized dual attention YOLO and FeatExProNet-based feature extraction. This involves capturing various features, such as shape (including area, major and minor axis length, perimeter, and solidity), color (utilizing histogram-based features for color distribution patterns), texture (employing Tamura and Haralick texture features for enhanced texture representation), statistical features (extracting moments like mean, skewness, and kurtosis for statistical characteristics), and deep learning-based features (leveraging Inception V3 for automatic and hierarchical feature extraction). The subsequent feature selection phase adopts a Hybrid Optimization Approach, combining Binary Sand Cat

Swarm Optimization and Butterfly Optimization algorithms to effectively select the most pertinent features. The concluding step integrates a VarioFusionNet-based model, seamlessly combining Vision Transformer, Google Net, Alex Net, DenseNet-121, ResNet-50, and Efficient Net for leaf disease detection. This holistic approach ensures not only the integration of diverse architectures but also a commitment to accuracy at every stage of the methodology.

Declarations:

Funding

On Behalf of all authors the corresponding author states that they did not receive any funds for this project.

Conflicts of Interest

The authors declare that we have no conflict of interest.

Competing Interests

The authors declare that we have no competing interest.

References

[1] Jiang, P., Chen, Y., Liu, B., He, D. and Liang, C., 2019. Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks. IEEE Access, 7, pp.59069-59080.

[2] Hu, G., Wu, H., Zhang, Y. and Wan, M., 2019. A low shot learning method for tea leaf's disease identification. Computers and Electronics in Agriculture, 163, p.104852.

[3] Liu, B., Zhang, Y., He, D. and Li, Y., 2017. Identification of apple leaf diseases based on deep convolutional neural networks. Symmetry, 10(1), p.11.

[4] Thomas, S., Kuska, M.T., Bohnenkamp, D., Brugger, A., Alisaac, E., Wahabzada, M., Behmann, J. and Mahlein, A.K., 2018. Benefits of hyperspectral imaging for plant disease detection and plant protection: a technical perspective. Journal of Plant Diseases and Protection, 125, pp.5-20.

[5] Ji, M., Zhang, L. and Wu, Q., 2020. Automatic grape leaf diseases identification via UnitedModel based on multiple convolutional neural networks. Information Processing in Agriculture, 7(3), pp.418-426.

[6] Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D. and Stefanovic, D., 2016. Deep neural networks-based recognition of plant diseases by leaf image classification. Computational intelligence and neuroscience, 2016.

[7] Ozguven, M.M. and Adem, K., 2019. Automatic detection and classification of leaf spot disease in sugar beet using deep learning algorithms. Physica A: statistical mechanics and its applications, 535, p.122537.

[8] Bai, X., Li, X., Fu, Z., Lv, X. and Zhang, L., 2017. A fuzzy clustering segmentation method based on neighborhood grayscale information for defining cucumber leaf spot disease images. Computers and Electronics in Agriculture, 136, pp.157-165.

[9] Zhang, X., Qiao, Y., Meng, F., Fan, C. and Zhang, M., 2018. Identification of maize leaf diseases using improved deep convolutional neural networks. Ieee Access, 6, pp.30370-30377.

[10] Singh, U.P., Chouhan, S.S., Jain, S. and Jain, S., 2019. Multilayer convolution neural network for the classification of mango leaves infected by anthracnose disease. IEEE access, 7, pp.43721-43729.

[11] Hassan, S.M., Maji, A.K., Jasiński, M., Leonowicz, Z. and Jasińska, E., 2021. Identification of plant-leaf diseases using CNN and transfer-learning approach. Electronics, 10(12), p.1388.

[12] Zhang, S., Zhang, S., Zhang, C., Wang, X. and Shi, Y., 2019. Cucumber leaf disease identification with global pooling dilated convolutional neural network. Computers and Electronics in Agriculture, 162, pp.422-430.

[13] Chemura, A., Mutanga, O. and Dube, T., 2017. Separability of coffee leaf rust infection levels with machine learning methods at Sentinel-2 MSI spectral resolutions. Precision Agriculture, 18, pp.859-881.

[14] Ma, J., Du, K., Zheng, F., Zhang, L., Gong, Z. and Sun, Z., 2018. A recognition method for cucumber diseases using leaf symptom images based on deep convolutional neural network. Computers and electronics in agriculture, 154, pp.18-24.

[15] Ramesh, S. and Vydeki, D., 2020. Recognition and classification of paddy leaf diseases using Optimized Deep Neural network with Jaya algorithm. Information processing in agriculture, 7(2), pp.249-260.

[16] Barbedo, J.G., 2018. Factors influencing the use of deep learning for plant disease recognition. Biosystems engineering, 172, pp.84-91.

[17] Panigrahi, K.P., Das, H., Sahoo, A.K. and Moharana, S.C., 2020. Maize leaf disease detection and classification using machine learning algorithms. In Progress in Computing, Analytics and Networking: Proceedings of ICCAN 2019 (pp. 659-669). Springer Singapore.

[18] Pantazi, X.E., Moshou, D. and Tamouridou, A.A., 2019. Automated leaf disease detection in different crop species through image features analysis and One Class Classifiers. Computers and electronics in agriculture, 156, pp.96-104.

[19] Singh, V. and Misra, A.K., 2017. Detection of plant leaf diseases using image segmentation and soft computing techniques. Information processing in Agriculture, 4(1), pp.41-49.

[20] Gonzalez-Huitron, V., León-Borges, J.A., Rodriguez-Mata, A.E., Amabilis-Sosa, L.E., Ramírez-Pereda, B. and Rodriguez, H., 2021. Disease detection in tomato leaves via CNN with lightweight architectures implemented in Raspberry Pi 4. Computers and Electronics in Agriculture, 181, p.105951.

[21] Dhingra, G., Kumar, V. and Joshi, H.D., 2019. A novel computer vision based neutrosophic approach for leaf disease identification and classification. Measurement, 135, pp.782-794.

[22] Sibiya, M. and Sumbwanyambe, M., 2019. A computational procedure for the recognition and classification of maize leaf diseases out of healthy leaves using convolutional neural networks. Agri Engineering, 1(1), pp.119-131.

[23] Wu, G., Fang, Y., Jiang, Q., Cui, M., Li, N., Ou, Y., Diao, Z. and Zhang, B., 2023. Early identification of strawberry leaves disease utilizing hyperspectral imaging combing with spectral features, multiple vegetation indices and textural features. Computers and Electronics in Agriculture, 204, p.107553.

[24] Geetharamani, G. and Pandian, A., 2019. Identification of plant leaf diseases using a nine-layer deep convolutional neural network. Computers & Electrical Engineering, 76, pp.323-338.

[25] Bajwa, S.G., Rupe, J.C. and Mason, J., 2017. Soybean disease monitoring with leaf reflectance. Remote Sensing, 9(2), p.127.

[26] Zhang, H., Sindagi, V. and Patel, V.M., 2019. Image de-raining using a conditional generative adversarial network. IEEE transactions on circuits and systems for video technology, 30(11), pp.3943-3956.

[27] Garg, B. and Sharma, G.K., 2016. A quality-aware Energy-scalable Gaussian Smoothing Filter for image processing applications. Microprocessors and Microsystems, 45, pp.1-9.

[28] Negi, S.S. and Bhandari, Y.S., 2014, May. A hybrid approach to image enhancement using contrast stretching on image sharpening and the analysis of various cases arising using histogram. In International conference on recent advances and innovations in engineering (ICRAIE-2014) (pp. 1-6). IEEE.

[29] Ren, S., He, K., Girshick, R. and Sun, J., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems, 28.

[30] Chen, J., Yuan, Z., Peng, J., Chen, L., Huang, H., Zhu, J., Liu, Y. and Li, H., 2020. DASNet: Dual attentive fully convolutional Siamese networks for change detection in high-resolution satellite images. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 14, pp.1194-1206.

[31] https://www.researchgate.net/publication/349717475_Performance_Evaluation_of_Deep_CNN-Based_Crack_Detection_and_Localization_Techniques_for_Concrete_Structures

[32] Yu, Z., Dong, Y., Cheng, J., Sun, M. and Su, F., 2022. Research on Face Recognition Classification Based on Improved GoogleNet. Security and Communication Networks, 2022, pp.1-6.

[33] Han, X., Zhong, Y., Cao, L. and Zhang, L., 2017. Pre-trained alexnet architecture with pyramid pooling and supervision for high spatial resolution remote sensing image scene classification. Remote Sensing, 9(8), p.848.

[34] https://www.mdpi.com/2075-1702/10/11/1002.