# ALR-AM-Net: Efficient fruit disease identification using Anisotropic diffusion with Multi modal feature fusion

**G. Sathya Priya[1] (Corresponding Author), Dr. M. Safish Mary[2]**

[1]Research Scholar, (Reg no: 21211282282010), Department of Computer Science, St. Xavier's College (Autonomous), Palayamkottai, Affiliated to Manonmaniam Sundaranar University, Abishekapatti, Tirunelveli, Tamil Nadu, India – 627012.

[2]Assistant Professor, Department of Computer Science, St. Xavier's College (Autonomous), Palayamkottai, Tirunelveli – 627002, Tamil Nadu, India.

Received: 17.09.2024 Revised: 18.11.2024 Accepted: 05.12.2024

**Abstract:**

Identifying fruit diseases is essential to prevent crop losses and reduce economic challenges for farmers and traders. Effective detection ensures quality standards, supports trade, and protects crops and ecosystems from disease spread. This study focuses on developing a fruit disease identification using datasets of Citrus, Guava, Papaya, and Apple fruits. These fruits are affected by various diseases. Citrus fruits are susceptible to blackspot, canker, screening, and scab. Guava fruits can be afflicted by Phytophthora, Root disease, and Scab. Papaya fruits face threats from Anthracnose, Black spot, Phytopthora, powdery mildew and ring spot. while Apple fruits are vulnerable to blotch, rot, and scab which can harm productivity and quality if not addressed early. The proposed system uses Enhanced Anisotropic Diffusion to preprocess images, enhancing important features and reducing noise. The ALR-AM-Net model combines tuned versions of AlexNet, LeNet, and ResNet with BiLSTM layers and an attention mechanism to focus on critical image details. Multi-level feature fusion used at the decision level ensures the model considers all extracted information for accurate classification. This approach enables early and accurate disease detection, reducing crop damage and improving agricultural sustainability. The results demonstrate that using Enhanced Anisotropic Diffusion with the ALR-AM-Net model achieves high accuracy, supporting better crop management and economic stability.

Keywords: Anisotropic Diffusion, ALR-AM-Net Model, Attention Mechanism, Fruit Disease Identification

## Introduction:

With the growing global population, food production has become one of the greatest challenges. It is estimated that food consumption will double by 2050, necessitating a more high-yielding and sustainable environment to increase plant yield [5].

India's diverse climate, soil, and land types greatly impact what crops and fruits farmers choose to grow, with their economic value being a key factor. Agriculture is a major part of a country's economy, as many nations depend on farming and related industries for income. Protecting and ensuring the safety of crops is essential for every country. In developing nations like India, challenges like malnutrition are closely tied to the need for better food security.

Pests and weeds are major factors that disrupt the natural growth of crops, reducing both quality and yield, and causing significant economic losses. Fruit trees play a vital role in a state's economy, while fruits are essential in human diets, providing key nutrients. With the global population rising, the demand for fruits and vegetables is growing rapidly. Early and accurate detection of fruit diseases is critical to prevent economic losses. Although experts can manually inspect and classify fruit diseases, this process is often time-consuming and expensive. Therefore, there is a strong need for smart, automated solutions. Since most infections appear visibly on fruits, computer vision-based methods are an effective way to detect diseases. To address the growing demand for food, it is imperative to integrate modern technology into the agriculture sector. In recent years, developing effective technology for agricultural monitoring has been challenging. Continuous monitoring of fruits is essential to take immediate precautions and prevent diseases from spreading to other parts of the fruit or adjacent fruits. Numerous studies, leveraging image processing, machine learning, and deep learning techniques, have significantly advanced the identification and recognition of fruit diseases.

Our study focuses on citrus, guava, papaya, and apple fruits. Among these, citrus plants stand out for their high vitamin C content and widespread popularity across the Indian subcontinent, the Middle East, and Africa. Citrus fruits provide numerous health benefits and are extensively used in the agricultural industry as raw materials for products like jams, sweets, ice creams, and confectionery. However, citrus plants are vulnerable to diseases such as blackspot, canker, greening, and scab [7].

Guava, a member of the Myrtaceae family, is an important fruit originally from the American tropics. It was introduced to Portugal in the early 17th century [8]. Guava is popular in both tropical and non-tropical regions, including countries like Bangladesh, India, Pakistan, Brazil, and Cuba. It is rich in nutrients like phosphorus, calcium, and nicotinic acid [9]. Guava also offers health benefits, such as controlling blood pressure, managing diabetes, boosting immunity against dysentery, and relieving diarrhea. It grows well in various soil types with pH levels between 4.4 and 4.9 and can survive extreme climate changes [10]. Common guava diseases include Phytophthora, root disease, and scab.

Apple is one of the most important tree fruits, ranking second in global fruit production [11]. In 2017, global apple production reached 83.1 million tons, highlighting its popularity worldwide [12]. Apples are versatile and can be eaten fresh or used in various products [13]. About 33% of apples are processed into items like juices, ciders, applesauce, dried apples, and other products [14]. However, the apple industry faces significant losses due to diseases that reduce product quality. Although visual inspection can identify diseased apples, human evaluation is often subjective and error-prone. Accurate and timely disease detection is essential to minimize losses. Apples are susceptible to diseases like blotch, rot, and scab. For our study, the dataset includes 16 classes, with 11,198 images of both healthy and diseased fruits, sourced from Kaggle.

Papaya is a major fruit crop in India, the world's largest producer, with an annual output of around 3 million tons. Despite its significance, research on papaya fruit quality classification remains limited [15]. Identifying healthy and diseased papayas accurately is vital for efficient marketing and export. Manual inspection often leads to errors, highlighting the need for an automated disease detection system. Common diseases affecting papayas include Anthracnose, Black Spot, Phytophthora, Powdery Mildew, and Ring Spot.

Our proposed ALR-AM-Net model integrates anisotropic diffusion with advanced deep learning architectures, such as BiAlexNet, BiLeNet, and BiResNet, along with an attention mechanism and multimodal feature fusion techniques. This approach improves the model's ability to accurately identify and classify fruit diseases, aiding in effective agricultural management and boosting economic outcomes.

**Article Organization:**

The article follows a structured approach, comprising different sections that address various aspects of the research. Section 2 provides an in-depth exploration of related works, offering a comprehensive overview of existing literature in the field. In Section 3, the methodology is presented in detail. Section 4 encompasses performance evaluation, result analysis, and comparison with earlier studies. Finally, Section 5 discusses conclusions. This structured approach ensures a systematic and organized presentation of the research, aiding in clear comprehension.

## Section - 2

**Related Work:**

The identification of fruit diseases has been extensively researched, with various machine learning and deep learning methods developed to enhance detection accuracy. Researchers have conducted numerous experiments using neural network models to classify and detect diseases from fruit images.

Vinay Kukreja et al. [16] aimed to improve the quality assessment of agricultural products to increase market value and reduce waste. They utilized a Convolutional Neural Network (CNN) to classify citrus fruits as either healthy or defective. Initially, their model achieved 67% accuracy on 150 images without preprocessing or data augmentation. However, after applying preprocessing and data augmentation on an expanded dataset of 1,200 images, the model's accuracy improved significantly to 89.1%. This study highlights the crucial role of preprocessing and data augmentation in enhancing the performance of deep learning models for fruit defect detection.

Asad Khattak et al. [17] proposed a CNN-based model to identify common citrus diseases, including black spot, canker, scab, greening, and melanose, in both fruits and leaves. The model uses multiple layers to extract key features and achieved a test accuracy of 94.55% on the Citrus and PlantVillage datasets. This accuracy surpasses other state-of-the-art models, offering a reliable solution for farmers to classify citrus diseases effectively.

Zongshuai Liu et al. [18] introduced a deep learning approach for citrus disease recognition using image analysis. They created a database of images for six common citrus diseases and utilized the MobileNetV2 model as the primary network. MobileNetV2 excelled in accuracy, speed, and compactness, achieving an 87.28% classification accuracy while reducing prediction time and model size. This innovation supports efficient, real-time disease detection on mobile devices, highlighting the potential of deep learning and mobile technology in managing agricultural diseases.

Walaa N. Jasim et al. [19] explored using Convolutional Neural Networks (CNNs) for identifying and classifying citrus diseases. They used a dataset of 2,450 images, covering seven disease types, including anthracnose, brown rot, and citrus canker. Their CNN model achieved an impressive 88% accuracy, showing its strong performance. They also found that the RMSProp optimization algorithm performed slightly better than ADAM. Additionally, the study highlighted the effectiveness of CNNs in agricultural applications, especially with the use of data augmentation.

Ashok Kumar Saini et al. [20] developed a deep learning system for early citrus disease detection in northern India, an important citrus-growing region. The system uses a CNN, enhanced with transfer learning and data augmentation, to analyze images of citrus leaves taken in the visible light spectrum. It categorizes diseases, pests, and nutritional issues into seven types. Integrated into an Android app, the system provides a quick, cost-effective alternative to traditional methods, achieving over 90% accuracy and minimal processing time, making it practical for field use.

Sukanya S. Gaikwad et al. [21] studied the classification of fruit leaf diseases using deep convolutional neural networks (CNNs), focusing on Apple, Custard Apple, and Guava leaves from the Hyderabad and Karnataka regions in India. The study aimed to address the gap in fungal classification by introducing a dataset of 14,181 images, spanning ten disease types, with variations in color, black and white, and grayscale images. They trained two CNN models, AlexNet and SqueezeNet, achieving recognition accuracies of 86.8% and 86.6% on color images, respectively. The results highlight the effectiveness of color images for disease classification and the potential for improving fungal classification with diverse dataset variations.

Hashan et al. [22] developed an advanced guava fruit disease detection system (GFDI) to improve production and reduce economic losses. The system uses an enhanced convolutional neural network (improved-CNN) based on AlexNet, incorporating techniques such as data augmentation, contrast enhancement, image resizing, and dataset splitting. Trained on 612 images, the improved-CNN achieved 98% accuracy on training data and 93% accuracy on test data with a learning rate of 0.001, while reducing model parameters by over 50 million compared to traditional AlexNet. This study highlights the model's effectiveness in improving disease detection accuracy and convergence rates, marking a significant advancement in guava disease management.

Abdur Nur Tusher et al. [23] looked at using deep learning to detect crop diseases in Bangladesh, where agriculture is a major part of the economy. Diseases like bacterial blight and leaf brown spot threaten crops such as guava, mango, rice, corn, and peach. Early detection is key to reducing crop damage, but many farmers still rely on manual methods, which are slow and often inaccurate. To solve this, the researchers developed an automated system using Convolutional Neural Networks (CNN) for disease detection. Their model, trained on a large dataset, achieved an impressive accuracy of 95.26%. This system can help farmers detect diseases faster and more accurately, improving crop management.

Piyush Kumar Pareek et al. [24] focused on using deep learning to detect diseases in grapefruit leaves. With grapefruits being an important crop worldwide, it's crucial to catch diseases early, but many farmers lack the skills to do so, leading to poor yields. The study proposed a system that first uses K-means clustering to isolate image backgrounds and then applies a CNN for disease classification. They also improved the model's accuracy by fine-tuning its settings using Firefly and cyclic randomization techniques. The CNN model achieved 95% accuracy, which is much better than previous models, which only reached 88-90%. This research shows how combining deep learning and image processing can greatly improve disease detection and crop management.

Lili Fu et al. [25] worked on improving the detection and management of apple leaf diseases, which are crucial for the global apple industry. They designed a Convolutional Neural Network (CNN) based on AlexNet to identify five specific apple leaf diseases. Their model includes advanced techniques like dilated convolution to extract large-scale features with fewer parameters, a parallel convolution module for multi-scale feature extraction, and shortcut connections to handle additional complexities. An attention mechanism enhances the model's focus on relevant features while minimizing background interference. Global pooling replaces fully connected layers to reduce parameters and preserve feature integrity. The model achieved an impressive 97.36% accuracy and is compact at just 5.87 MB, offering a lightweight and effective solution for apple leaf disease identification.

## Section - 3

## Methodology

This study introduces the ALR-AM-Net model for precise fruit disease identification. The proposed workflow is illustrated in Figure 1. The process is organized into several key steps, each of which is detailed below.

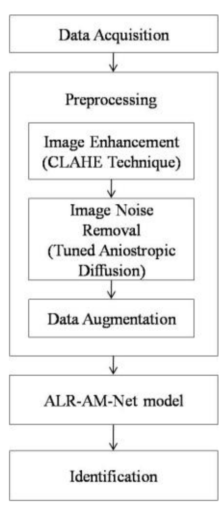



Fig 1: General flow diagram of the proposed ALR-AM-NET model

**Data Acquisition:**

In this study, a total of 12,838 images were utilized sourced from Kaggle [26,27,28,29], encompassing a variety of fruits and their conditions. The dataset includes both healthy and diseased fruits of citrus, guava, papaya and apple. The classification for each fruit diseases is as follows:

- **Citrus**: Categories include black spot, canker, greening, and scab.
- **Guava**: Disease categories consist of Phytophthora, root rot, and scab.
- **Papaya**: Diseases are classified into Anthracnose, black spot, phytopthora, powdery mildew and ring spot.
- **Apple**: Images are categorized into blotch, rot, normal and scab.

As outlined in Table 1, the dataset is split for training and testing purposes, with 80% of the data allocated for model training and 20% reserved for testing. This approach ensures that the model is evaluated on a representative subset of the data, facilitating an accurate assessment of its performance.

**Table 1**
The total number of samples for each class in the citrus and guava fruit disease dataset (After augmentation).

| **Class** | **Number of Samples** |
|---|---|
| **Apple** | |
| Blotch | 1280 |
| Normal | 800 |
| Rot | 1460 |
| Scab | 1000 |
| Total Images | 4540 |
| **Citrus** | |
| Blackspot | 1230 |
| Canker | 1023 |
| Greening | 160 |
| Scab | 241 |

| | |
|---|---|
| Healthy | 416 |
| Total Images | 3070 |
| **Guava** | |
| Phytopthora | 1520 |
| Root | 310 |
| Scab | 1240 |
| Total Images | 3070 |
| **Papaya** | |
| Anthracnose | 380 |
| Black spot | 340 |
| Healthy | 288 |
| Phytopthora | 310 |
| Powdery mildew | 310 |
| Ring spot | 530 |
| Total Images | 2158 |

**Preprocessing:**

Preprocessing encompasses Image Enhancement, Noise Removal and Data augmentation

**Image Enhancement:**

To enhance fruit disease images, the Contrast Limited Adaptive Histogram Equalization (CLAHE) method is utilized. CLAHE is designed to improve image contrast, making intricate details more discernible and significantly boosting overall image quality. This technique is particularly effective for images captured under non-uniform illumination or varying light conditions, where traditional methods might struggle. Unlike standard histogram equalization, CLAHE limits noise amplification during the contrast enhancement process, thereby preserving image quality and preventing the introduction of unwanted artifacts. The adaptive nature of CLAHE allows for localized contrast adjustments, meaning that different regions of an image can be enhanced independently based on their specific lighting conditions. This targeted enhancement ensures that the natural appearance of the image is maintained, while critical details are highlighted, making it easier to analyze and interpret. This is especially valuable in applications where clear visibility and high contrast are essential for accurate disease diagnosis and analysis in fruit images.

**Noise Removal:**

To remove noise from the fruit disease data, anisotropic diffusion is applied. This sophisticated image processing technique is designed to reduce noise while preserving crucial features like edges. Anisotropic diffusion smooths out noise without blurring important details, ensuring that key information, such as edges, remains intact. Here's process and the parameters used in anisotropic diffusion:

- In the anisotropic diffusion process, gradients are computed in four directions—north, south, east, and west—using the np.roll function to measure intensity differences between each pixel and its neighbors. The diffusion coefficients, which control the amount of smoothing, are calculated based on the chosen formula: exponential decay of squared gradients (Option 1), inverse quadratic function of gradients (Option 2), or exponential decay of absolute gradients (Option 3).
- A dynamic threshold is set, derived from the mean gradient magnitude, to adapt the diffusion process to the image's characteristics and regulate smoothing. The image is updated iteratively by adjusting each pixel's intensity according to the weighted sum of neighboring gradients, influenced by the diffusion coefficients. After each iteration, the algorithm checks if the mean diffusion value falls below the threshold; if so, the process halts early, signaling that the image has converged and further iterations would yield minimal changes.

**Parameters Used in Anisotropic Diffusion:**

In this anisotropic diffusion process, **20 iterations** are used for smoothing the image. The **kappa** value of **50** balances edge sensitivity, allowing more smoothing while preserving edges. The **gamma** parameter, set at **0.2**, controls the diffusion rate per iteration, ensuring moderate smoothing. The **option** parameter **1** employs an exponential function to calculate diffusion coefficients, effectively balancing noise reduction and edge preservation. The **delta_thresh** of **1e-4** serves as a convergence criterion, stopping the process when changes between iterations are minimal.

**Fig:2 Sample images of citrus, guava, papaya and apple fruit disease dataset**

### Data Augmentation:

Data augmentation is a powerful technique for enhancing the diversity of a dataset by applying various transformations to existing data. Although it doesn't create new data points, it significantly improves the performance and generalizability of deep learning models. By generating augmented versions of input data, techniques such as random rotations, shifts, flips, shearing, and zooming introduce variability that helps the model learn better. The fill_mode parameter determines how newly generated pixels are filled, with options like 'nearest,' which uses neighboring pixel values. Additional techniques such as centering, normalizing, and Zero-phase Component Analysis (ZCA) whitening can also be employed to further improve data quality and model training.

## Proposed model

The ALR-AM-NET model is designed by integrating hybrid architectures, where tuned versions of AlexNet, LeNet, and ResNet are combined with BiLSTM and it is named as BiAlexNet, BiLeNet, and BiResNet. An attention mechanism is applied to extract significant features, ensuring that key information is highlighted during the learning process. Following this, multilevel feature fusion is performed, which merges features at different levels, and the final decision is made by fusing these features at the decision level for more accurate classification results.

### Layers used for BiAlexNet:

This model extends the classic AlexNet by incorporating five convolutional layers. The architecture starts with a convolutional layer that applies 16 filters of size 3x3 with a stride of 1 and 'same' padding, followed by a max-pooling layer with a 2x2 window. This is followed by batch normalization. The second convolutional layer uses 32 filters of size 3x3, followed by another max-pooling layer (2x2) and batch normalization. The third convolutional layer applies 64 filters (3x3), with max-pooling (2x2) and batch normalization. The fourth layer uses 124 filters (3x3), followed by max-pooling (1x1) and batch normalization. The final convolutional layer applies 184 filters (3x3), with max-pooling (1x1) and batch normalization. An attention mechanism is used to highlight critical features, followed by a flattening layer. This is then processed by two BiLSTM layers with 150 and 200 units, respectively. The output is passed through two fully connected dense layers with 4096 and 3096 units, respectively, using ReLU activation. Dropout is applied with a rate of 0.2 to prevent overfitting, and the final output is produced by a softmax layer for classification.

### Layers used for BiLeNet

This model extends the classic LeNet architecture by incorporating three convolutional layers. The architecture begins with a convolutional layer that applies 16 filters of size 5x5 with ReLU activation, followed by an average pooling layer with a 2x2 window. The second convolutional layer uses 32 filters of size 5x5, followed by another average pooling layer (2x2). The third convolutional layer applies 64 filters of size 5x5, followed by average pooling with a 1x1 window. An attention mechanism is then used to emphasize key features, followed by a flattening layer. The flattened output is processed by two Bidirectional LSTM (BiLSTM) layers with 150 and 200 units, respectively. The output is then passed through two fully connected dense layers with 4096 and 3096 units, respectively, using ReLU activation. Dropout is applied with a rate of 0.5 to prevent overfitting, and the final output is produced by a softmax layer for classification.

**Layers used for BiResNet**

This model extends the classic ResNet architecture by incorporating five convolutional layers with residual connections. The architecture starts with a convolutional layer that applies 16 filters of size 3x3 with ReLU activation and batch normalization, followed by max-pooling with a 2x2 window. The second convolutional layer uses 32 filters of size 3x3, followed by batch normalization, max-pooling with a 2x2 window, and additional batch normalization. The third convolutional layer applies 64 filters of size 3x3, with batch normalization, max-pooling (2x2), and more batch normalization. The fourth layer uses 128 filters of size 3x3, followed by batch normalization, max-pooling (2x2), and additional batch normalization. The final convolutional layer applies 184 filters of size 3x3, with batch normalization and max-pooling (1x1). An attention mechanism is utilized to highlight important features, followed by a flattening layer. The flattened output is processed by two Bidirectional LSTM (BiLSTM) layers with 150 and 200 units, respectively. The output is then passed through two fully connected dense layers with 4096 and 3096 units, respectively, using ReLU activation. Dropout is applied with a rate of 0.5 to prevent overfitting, and the final output is produced by a softmax layer for classification.

In the ALR-AM-NET model, multi-level feature fusion is strategically employed to enhance classification accuracy and robustness. The process begins with extracting features from three distinct architectures: BiAlexNet, BiLeNet, and BiResNet. Each of these models uses a tuned alexnet, lenet, restnet and attention mechanism is used to highlight and prioritize critical features, ensuring that key information is effectively captured. Once these features are extracted, they are flattened and concatenated, creating a unified and comprehensive representation of the input data. The concatenated features are then processed through two Bidirectional LSTM layers. These LSTM layers are designed to capture complex sequential dependencies and patterns within the feature data, further enriching the feature representation. The attention mechanism applied earlier aids in emphasizing the most relevant features during this process. Following LSTM processing, the model passes the output through two fully connected dense layers. These layers refine and consolidate the information from the LSTM layers, improving the model's ability to make accurate predictions. Dropout (with rates of 0.2 or 0.5) and batch normalization are applied to these dense layers to prevent overfitting and ensure better generalization of the model.

Finally, the model outputs class probabilities through a softmax layer, providing the final classification result. The multi-level feature fusion approach integrates features extracted at different levels of abstraction and combines them at the decision level, allowing the ALR-AM-NET model to leverage the strengths of each architecture. This comprehensive feature integration, enhanced by the attention mechanism, contributes to the model's improved accuracy and robustness.

**Fig: 3 Proposed ALR-AM-Net model with Attention mechanism based on multi modal feature fusion**

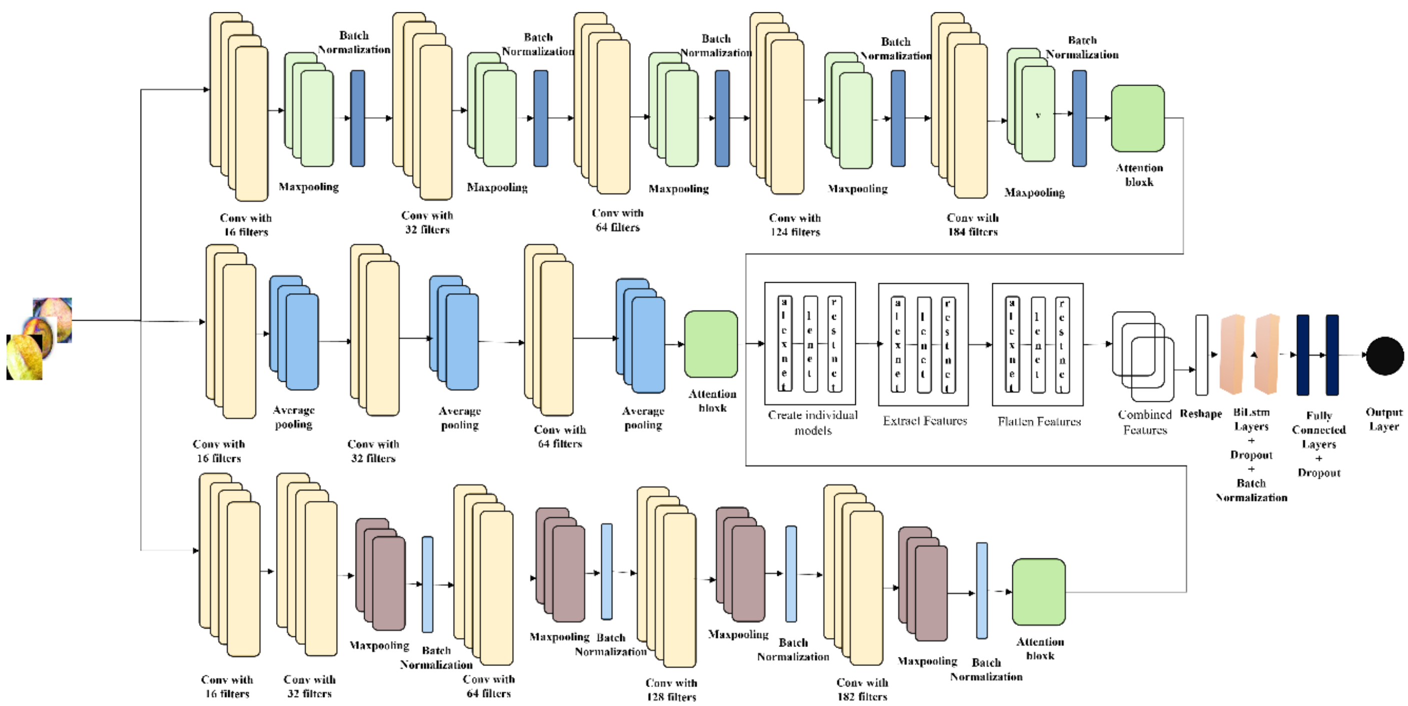



**Table 2: Layers and parameters for the proposed ALR-AM-Net model**

| Layer Name | Kernel Size/Pool Size | Stride | Padding | Output channel |
|---|---|---|---|---|
| Conv2D | 3*3 | 1*1 | Same | 16 |
| Maxpooling 2D | 2*2 | - | - | - |
| Conv2D | 3*3 | 1*1 | Same | 32 |
| Maxpooling 2D | 2*2 | - | - | - |
| Conv2D | 3*3 | 1*1 | Same | 64 |
| Maxpooling 2D | 2*2 | - | - | - |
| Conv2D | 3*3 | 1*1 | Same | 122 |
| Maxpooling 2D | 2*2 | - | - | - |
| Conv2D | 3*3 | 1*1 | Same | 184 |
| Maxpooling 2D | 2*2 | - | - | - |

| | | | | |
|---|---|---|---|---|
| **Conv2D** | **5*5** | **-** | **-** | **16** |
| **AveragePooling2D** | **2*2** | | | |
| **Conv2D** | **5*5** | **-** | **-** | **32** |
| **AveragePooling2D** | **2*2** | | | |
| **Conv2D** | **5*5** | **-** | **-** | **64** |
| **AveragePooling2D** | **1*1** | **-** | **-** | **-** |
| **Conv2D** | **3*3** | **1*1** | **Same** | **16** |
| **Conv2D** | **3*3** | **1*1** | **Same** | **32** |
| **Maxpooling 2D** | **2*2** | **-** | **-** | |
| **Conv2D** | **3*3** | **1*1** | **Same** | **64** |
| **Maxpooling 2D** | **2*2** | **-** | **-** | |
| **Conv2D** | **3*3** | **1*1** | **Same** | **128** |
| **Maxpooling2D** | **2*2** | **-** | **-** | |
| **Conv2D** | **3*3** | **1*1** | **Same** | **182** |
| **Maxpooling2D** | **2*2** | **-** | **-** | **-** |
| **Bidirectional** | **-** | **-** | **-** | **150** |
| **Bidirectional** | **-** | **-** | **-** | **200** |
| **Dense** | **-** | **-** | **-** | **4096** |
| **Dense** | **-** | **-** | **-** | **3096** |
| **Dense** | **-** | **-** | **-** | **5** |

**Each block utilizes the batch normalization after the maxpooling layer**

**Table 4: Training Parameters**

| Optimizer | Epoch | Batch Size | Learning Rate | Weight_decay | Beta_1 | Beta-2 |
|---|---|---|---|---|---|---|
| Adam | 300 | 32 | 0.00001 | 0.0001 | 0.9 | 0.999 |

**Table 5: Callback: Reduce Learning Rate on Plateau**

| Parameter | Monitor | patience | Factor | min_lr |
|---|---|---|---|---|
| Value | val_accuracy | 5 | 0.2 | 0.0001 |

**Mathematical representation of the ALR-AM-NET model:**

The ALR-AM-NET model integrates convolutional neural networks (CNNs) such as AlexNet, LeNet, and ResNet, along with Bidirectional LSTM (BiLSTM) and attention mechanisms to form a hybrid architecture. This model also uses multi-level feature fusion and decision-level fusion to improve accuracy in disease identification tasks. Below is a mathematical representation of the components and the entire model.

**BiAlexNet layer representation:**

For an input image, $X_{alexnet} \in R^{H*W*C}$ AlexNet consists of several convolutional, max-pooling, and normalization layers. For each convolutional layer, the operation is defined as:

$A^{l}_{alexnet}$= ReLU($W^{-l}_{alexnet} * A^{l-1}_{alexnet} + b^{l}_{alexnet}$)

where

* represents the convolution operation.
$w^{l}_{alexnet}$ is the weight matrix at layer l.
$A^{l-1}_{alexnet}$ is the input from the previous layer.
$b^{l}_{alexnet}$ is the bias.
ReLU is the activation function.

**Each convolution layer is followed by max-pooling and batch normalization:**

$P^{l}_{alexnet}$=MaxPooling($A^{l}_{alexnet}$), $N^{l}_{alexnet}$=BatchNorm($P^{l}_{alexnet}$)

The output of the final AlexNet convolution layer is flattened:

$F_{alexnet}$=Flatten($N^{l}_{alexnet}$)

**2. BiLeNet Layer Representation:**

For an input image $X_{lenet} \in R^{H*W*C}$, LeNet is defined similarly with convolutional and pooling layers:

$A^{l}_{lenet}$=ReLU($W^{l}_{lenet}$*$A^{l-1}_{lenet}$+$b^{l}_{lenet}$)

$P^{l}_{lenet}$=AvgPooling($A^{l}_{lenet}$)

Flatten the final LeNet output:

$F_{lenet}$=Flatten($P^{L}_{lenet}$)

**3. BiRestNet Layer Representation:**

ResNet uses residual blocks with skip connections. The residual block operation is

$R^{l}_{resnet}$=ReLU($W^{l}_{resnet}$*$A^{l-1}_{resnet}$+$b^{l}_{resnet}$)+$A^{l-1}_{resnet}$

Here, $R^{l}_{resnet}$ is the residual output of the block, where the input is added to the output of the convolution layers.

The final output of ResNet is:

$F_{resnet}$+=Flatten($R^{L}_{resnet}$)

**4. Feature Fusion:**

After extracting the features from AlexNet, LeNet and Resnet, the features are concatenated:

$F_{combined}$=Concatenate($F_{alexnet,}F_{lenet,}F_{resnet}$)

**5. Attention Mechanism:**

The attention mechanism is applied to the concatenated features $F_{combined}$. The attention weights α are computed as:

$e_i$=tanh($W_{att}F_{combined}$+$b_{att}$)

$\alpha_i=\frac{exp(e_i)}{\sum_j exp(e_j)}$

The output after attention is:

$F_{att}$=$\sum \alpha_i F_{combined}$

i

**6. BiLSTM Layers:**

The output from the attention mechanism $F_{att}$ is fed into two bidirectional LSTM layers, for each LSTM layer, the forward and backward LSTMs are calculated as:

$h_t$ = LSTM($F_{att}$,$h_{t-1}$)

$h_t$ = LSTM($F_{att}$,$h_{t+1}$)

The final hidden state is:

$H_t$=Concatenate($ht$,$h_t$)

This process is repeated for both BiLSTM layers, resulting in

$H_{bilstm}$ = $H_{bilstm2}$($H_{bilstm1}$($F_{att}$))

**7. Fully Connected Layers:**

The BiLSTM output is passed through dense (fully connected) layers:

$D_1 = ReLU(W_1H_{bilstm} + b1)$

$D_2 = ReLU(W_2D_1 + b1)$

**8. Classification output:**

The final layer is a softmax classifier for multi-class output:

$y = Softmax(W_3D_2 + b_3)$

The softmax operation ensures that the output is a probability distribution over C classes:

$y_i = \frac{exp(y_{i)}}{\sum_{j=1}^{C} exp(y_{i)}}$

**Complete Model Equation:**

The overall function of the ALR-AM-NET model is:

$y = Softmax(W_3ReLU(W_2ReLU(W_1H_{bilstm} + b_1) + b_2) + b_3)$

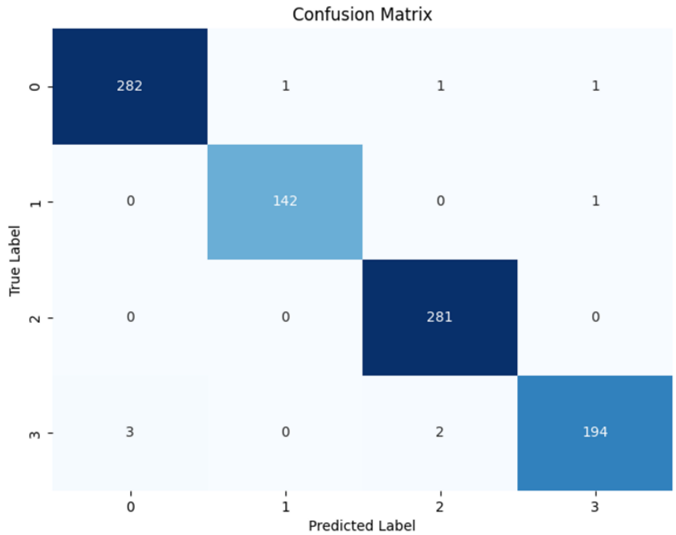
Where $H_{bilstm}$ is the output of the BiLSTM layers applied to the attention-modified combined features $F_{att.}$ This summarizes the mathematical flow of the ALR-AM-NET model, covering convolutional layers, feature fusion, attention mechanism, BiLSTM and fully connected layer leading to classification.

## Section - 4

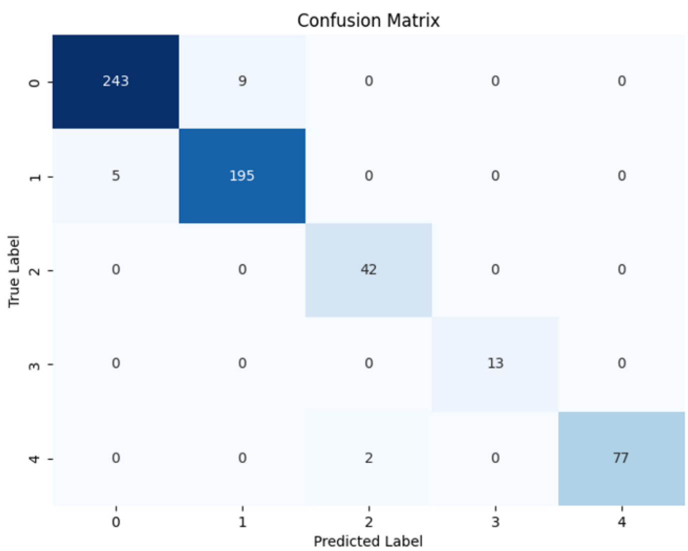
**Results and Discussion:**

The experimental assessment of the ALR-AM-NET model relies on the specified architectural and training parameters. The algorithm's performance is evaluated using test data on a laptop PC featuring an Intel(R) Core (TM) i3-6100U CPU @ 2.30GHz, with 4.00 GB of RAM. The deep learning model and result calculations are executed within the Python Idle environment. The training time for the ALR-AM-NET model varies by dataset size, with a maximum of 38 seconds per epoch for papaya fruit disease identification, 60 seconds for apple, 45 seconds for citrus, and 41 seconds for guava. The model is trained with a batch size of 32 and 300 epochs. The training and test accuracy and loss graphs for the ALR-AM-NET model are illustrated in Figures 10 and 11. Both training and test losses gradually converge towards a minimum, indicating effective learning and model generalization by the conclusion of the graph. The resulting confusion matrix, shown in Figure 12, provides a graphical depiction of the classification accuracy for each category. Additionally, performance indicators such as F1-score, accuracy, recall, and precision are calculated. These metrics offer a comprehensive evaluation of the model's performance in classification tasks and illustrate its ability to recognize and differentiate between various classes.

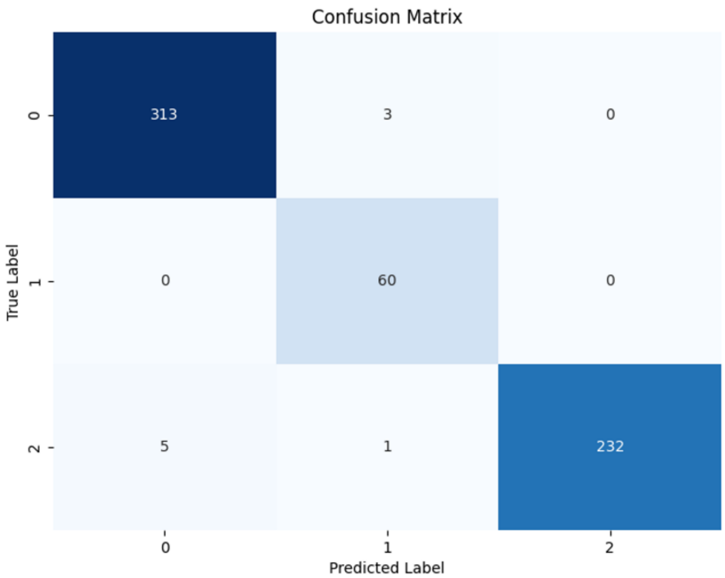
**Fig 4,5,6: Confusion matrix for ALR-AM-Net model with Attention mechanism based on multi modal feature fusion**



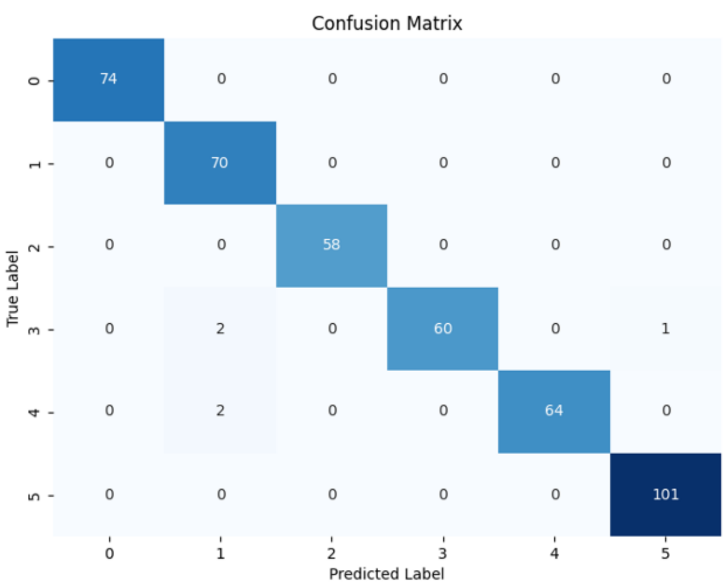

Apple




Citrus

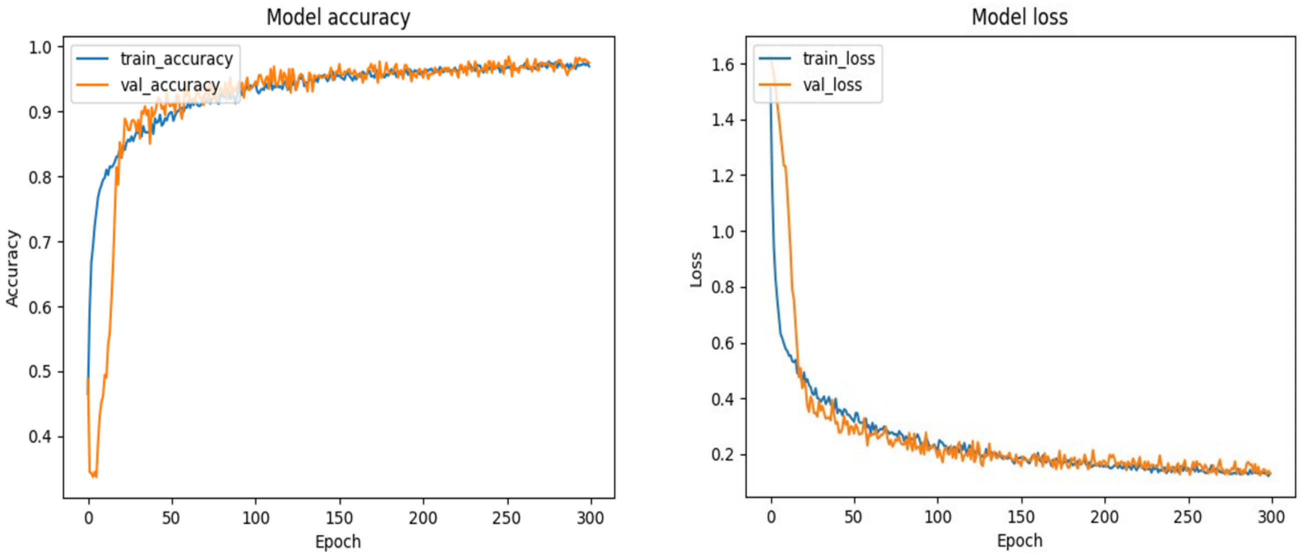



Guava

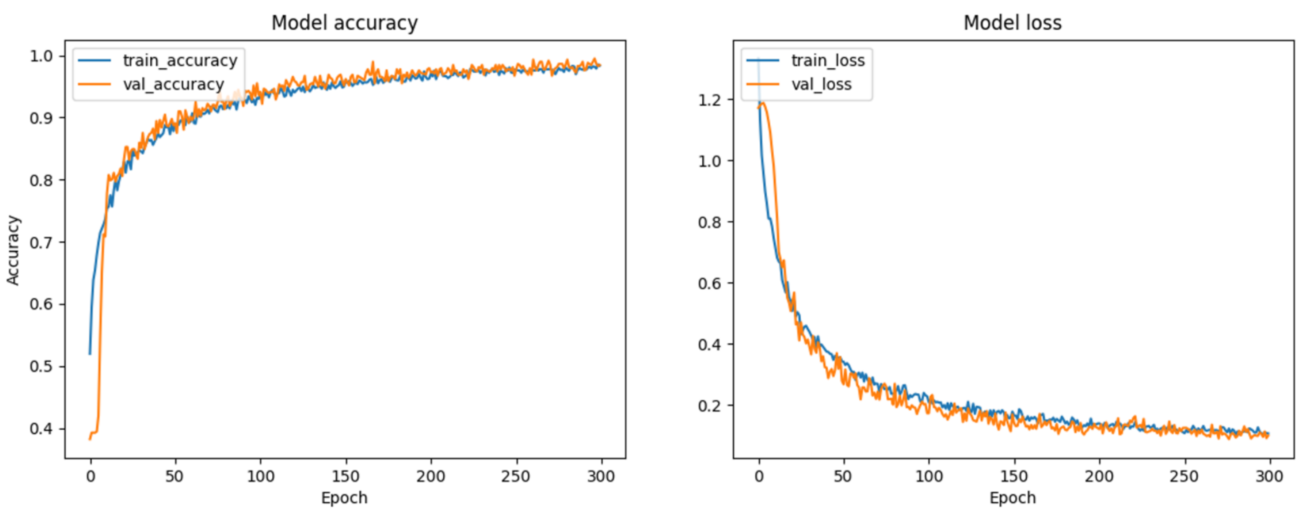



Papaya

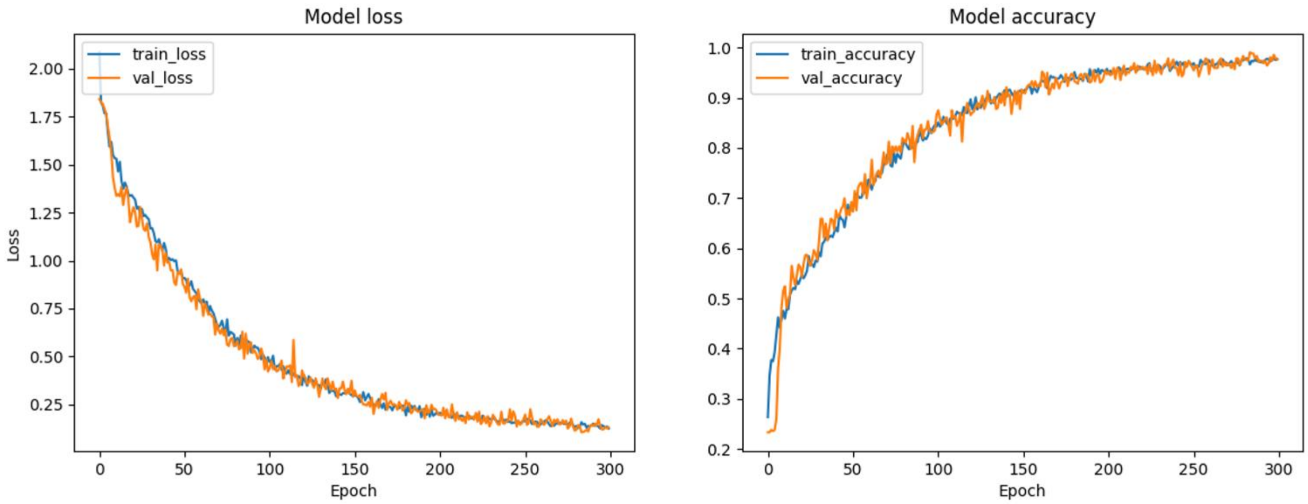
Citrus




Guava



Papaya




Apple

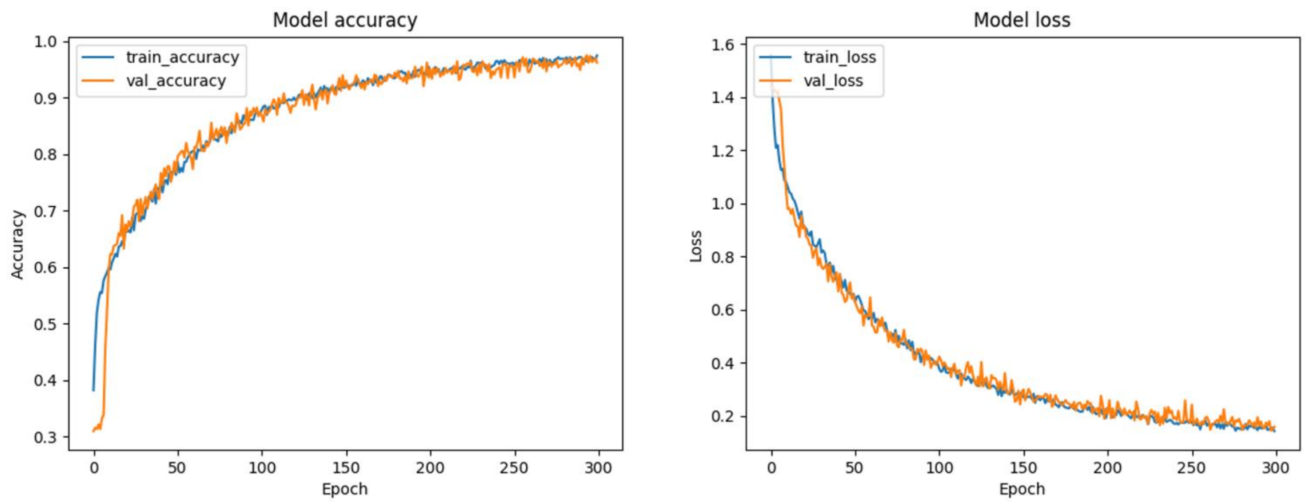
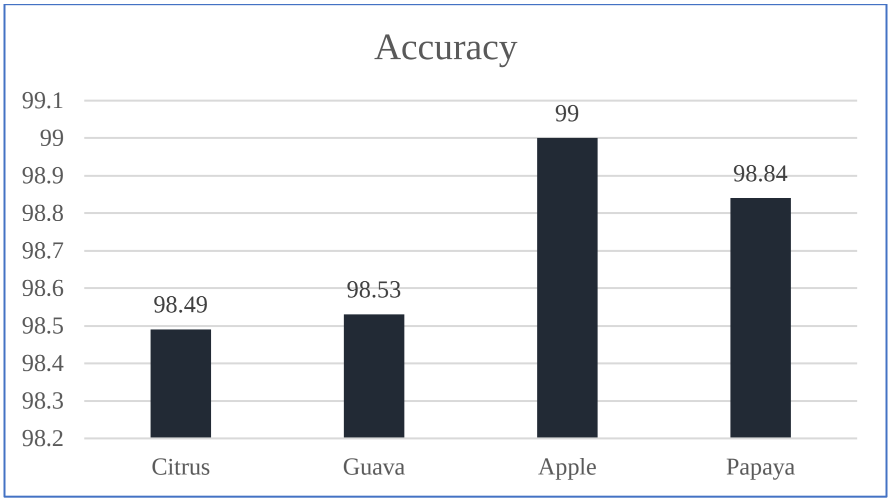



**Table 6: Predicted results of the proposed ALR-AM-Net model**

| Dataset | Accuracy | Precision | F1-Score | Specificity | Sensitivity |
|---|---|---|---|---|---|
| Citrus | 98.49 | 99.03 | 98.76 | 98.88 | 99.56 |
| Guava | 98.53 | 97.39 | 98.84 | 98.07 | 99.20 |
| Apple | 99.00 | 99.04 | 98.93 | 98.98 | 99.65 |
| Papaya | 98.84 | 98.93 | 98.70 | 98.79 | 99.76 |

**Fig 7: Proposed accuracy results for various dataset**




**Comparison between the proposed approach and previous research:**

In the area of fruit disease identification, various studies have employed distinct methods, each showcasing its own level of accuracy. The fundamental objective of this research is to demonstrate the efficiency of the ALR-AM-Net model in accurately identifying fruit diseases. Unlike previous research based on deep learning, our study stands out due to its methodological contributions.

In Below, we compare our study with previous works that employed different deep learning methods, focusing on accuracy as a key metric. The results demonstrate the efficacy of our proposed approach achieves comparable accuracy to previous studies and it shows effectiveness in fruit disease identification. Notable studies and their respective methods, along with the achieved accuracies, are outlined below

1. Monali Parmar [30]: Leveraged Deep Learning techniques identifying papaya disease using vision transformers with the accuracy of 91%.
2. Xulu Gong et al. [31] utilized the improved Faster R-CNN for finding the apple leaf Disease with 63.01% accuracy.
3. Piyush Kumar Pareek et al [32] proposed the 1-d CNN for grape fruit disease identification with the accuracy of 95%.
4. Khandakar Rabbi Ahmed et al [33] employs transfer learning approach for guava fruit disease identification with 74.02%
5. Varun Kumar et al [34] employs a CNN and random forest for papaya leaf disease identification wth an accuracy of 94.49%.
6. Radhika Gupta et al. [35]: Utilized lemon for classify various lemon disorders using hybrid model that incorporates Support Vector Machines (SVM) and Convolutional Neural Networks (CNN)resulting in an accuracy of 89.6%.

## Section - 5

### Conclusion:

This study addresses the critical need for effective fruit disease identification to prevent crop losses and economic burdens. By utilizing datasets of Citrus, Guava, Papaya, and Apple fruits, we have developed a robust identification system designed to address the diverse range of diseases affecting these fruits. Our approach combines Enhanced Anisotropic Diffusion with the advanced ALR-AM-NET model, which preprocesses images to emphasize critical features and reduce noise, ensuring precise identification. The ALR-AM-NET model integrates BiAlexNet, BiLeNet, BiRestNet and an attention mechanism, enhancing the model's ability to focus on relevant features for accurate classification. Multi-level feature fusion, utilizing late fusion to combine features from various layers and modalities at the decision level, further refines the model, ensuring comprehensive and reliable disease identification. This research advances fruit disease detection, aiming to reduce disease spread and improve crop management. The proposed method offers practical benefits for stakeholders, supporting better crop health and productivity.

### References:

1. Indian Agriculture Sector, Farming in India | IBEF'. India Brand Equity Foundation, https://www.ibef.org/industry/agriculture-india. Accessed 25 July 2024
2. Gavhale KR, Gawande U (2014) An overview of the research on plant leaves disease detection using image processing techniques. IOSR J Comput Eng (IOSR-JCE) 16(1):10–16
3. Samajpati BJ, Degadwala SD (2016) Hybrid approach for apple fruit diseases detection and classification using random forest classifier. In: 2016 international conference on communication and signal processing (ICCSP). IEEE, pp 1015–1019
4. Camargo A, Smith JS (2009) An image-processing based algorithm to automatically identify plant disease visual symptoms. Biosys Eng 102(1):9–21
5. Hunter, Mitchell C., et al. "Agriculture in 2050: Recalibrating Targets for Sustainable Intensification." *BioScience*, vol. 67, no. 4, Apr. 2017, pp. 386–91. *DOI.org (Crossref)*, https://doi.org/10.1093/biosci/bix010.
6. R. Manavalan, "Automatic identification of diseases in grains crops through computational approaches: A review", *Comput. Electron. Agricult.*, vol. 178, Nov. 2020.
7. W. Pan, J. Qin, X. Xiang, Y. Wu, Y. Tan and L. Xiang, "A smart mobile diagnosis system for citrus diseases based on densely connected convolutional networks", *IEEE Access*, vol. 7, pp. 87534-87542, 2019.
8. Bose, T. K. Ed. *Fruits Of India - Tropical And Subtropical*. *Internet Archive*, http://archive.org/details/in.ernet.dli.2015.461052. Accessed 30 July 2024.
9. Farhan Al Haque, A. S. M., et al. "A Computer Vision System for Guava Disease Detection and Recommend Curative Solution Using Deep Learning Approach." *2019 22nd International Conference on Computer and Information Technology (ICCIT)*, IEEE, 2019, pp. 1–6. *DOI.org (Crossref)*, https://doi.org/10.1109/ICCIT48885.2019.9038598.

10. Keith, Lisa M., et al. "Identification and Characterization of *Pestalotiopsis* Spp. Causing Scab Disease of Guava, *Psidium Guajava* , in Hawaii." *Plant Disease*, vol. 90, no. 1, Jan. 2006, pp. 16–23. *DOI.org (Crossref)*, https://doi.org/10.1094/PD-90-0016.
11. Musacchi, Stefano, and Sara Serra. "Apple Fruit Quality: Overview on Pre-Harvest Factors." *Scientia Horticulturae*, vol. 234, Apr. 2018, pp. 409–30. *DOI.org (Crossref)*, https://doi.org/10.1016/j.scienta.2017.12.057.
12. Cebulj, Anka, et al. "Importance of Metabolite Distribution in Apple Fruit." *Scientia Horticulturae*, vol. 214, Jan. 2017, pp. 214–20. *DOI.org (Crossref)*, https://doi.org/10.1016/j.scienta.2016.11.048.
13. Faostat, F. Food and agriculture organization of the United Nations (FAO) 2017. Available online: http://www.fao.org/faostat/en/-data/QC (accessed on 2 April 2021).
14. Skinner, R. Chris, et al. "A Comprehensive Analysis of the Composition, Health Benefits, and Safety of Apple Pomace." *Nutrition Reviews*, Aug. 2018. *DOI.org (Crossref)*, https://doi.org/10.1093/nutrit/nuy033.
15. *PAPAYA*. https://nhb.gov.in/Horticulture%20Crops/Papaya/Papaya1.htm. Accessed 5 Sept. 2024.
16. [IEEE 2020 International Conference on Smart Electronics and Communication (ICOSEC) - Trichy, India (2020.9.10-2020.9.12)] 2020 International Conference on Smart Electronics and Communication (ICOSEC) - A Deep Neural Network Based Disease Detection Scheme for Citrus Fruits | 10.1109/ICOSEC49089.2020.9215359-Sci_hub. https://www.wellesu.com/https://ieeexplore.ieee.org/abstract/document/9215359. Accessed 7 Aug. 2024.
17. Khattak, Asad, et al. 'Automatic Detection of Citrus Fruit and Leaves Diseases Using Deep Neural Network Model'. IEEE Access, vol. 9, 2021, pp. 112942–54. IEEE Xplore, https://doi.org/10.1109/ACCESS.2021.3096895.
18. Liu, Zongshuai & Xiang, Xuyu & Qin, Jiaohua & YunTan, & Zhang, Qin & Xiong, Naixue. (2020). Image Recognition of Citrus Diseases Based on Deep Learning. Computers, Materials & Continua. 66. 457-466. 10.32604/cmc.2020.012165.
19. Jasim, W. N., Sead Almola, S. A., Alabiech, M. H., & Harfash, E. A. J. (2022). Citrus Diseases Recognition by Using CNN Model. *Informatica (03505596)*, *46*(7).
20. Saini, A. K., Bhatnagar, R., & Srivastava, D. K. (2021). AI based automatic detection of citrus fruit and leaves diseases using deep neural network model. *Journal of Discrete Mathematical Sciences and Cryptography*, *24*(8), 2181–2193. https://doi.org/10.1080/09720529.2021.2011095
21. Gaikwad, Sukanya S., et al. 'Fungi Affected Fruit Leaf Disease Classification Using Deep CNN Architecture'. International Journal of Information Technology, vol. 14, no. 7, Dec. 2022, pp. 3815–24. Springer Link, https://doi.org/10.1007/s41870-022-00860-w.
22. Mahamudul Hashan, Antor, et al. 'Guava Fruit Disease Identification Based on Improved Convolutional Neural Network'. International Journal of Electrical and Computer Engineering (IJECE), vol. 14, no. 2, Apr. 2024, p. 1544. DOI.org (Crossref), https://doi.org/10.11591/ijece.v14i2.pp1544-1551.
23. Tusher, Abdur Nur, et al. 'Automatic Recognition of Plant Leaf Diseases Using Deep Learning (Multilayer CNN) and Image Processing'. Third International Conference on Image Processing and

Capsule Networks, edited by Joy Iong-Zong Chen et al., Springer International Publishing, 2022, pp. 130–42. Springer Link, https://doi.org/10.1007/978-3-031-12413-6_11.

24. Pareek, Piyush Kumar, et al. ‘Clustering Based Segmentation with 1D-CNN Model for Grape Fruit Disease Detection’. 2023 IEEE International Conference on Integrated Circuits and Communication Systems (ICICACS), IEEE, 2023, pp. 1–7. DOI.org (Crossref), https://doi.org/10.1109/ICICACS57338.2023.10099916.
25. Fu, Lili, et al. ‘Lightweight-Convolutional Neural Network for Apple Leaf Disease Identification’. Frontiers in Plant Science, vol. 13, May 2022. Frontiers, https://doi.org/10.3389/fpls.2022.831219.
26. Apple Fruit Disease. https://www.kaggle.com/datasets/kaivalyashah/apple-disease-detection. Accessed 14 Aug. 2024.
27. Research Data - Mendeley Data. https://data.mendeley.com/research-data/? Accessed 27 Jan. 2024.
28. Guava Fruit disease images. (2023, May 24). Kaggle. https://www.kaggle.com/datasets/mhantor/guava fruit-disease
29. Pomegranate Fruit Diseases [Image] Dataset. https://www.kaggle.com/datasets/sujaykapadnis/pomegranate-fruit-diseases-dataset. Accessed 14 Aug. 2024.
30. Parmar, Monali, and Dr Sheshang Degadwala. “Deep Learning for Accurate Papaya Disease Identification Using Vision Transformers.” International Journal of Scientific Research in Computer Science, Engineering and Information Technology, vol. 10, no. 2, Mar. 2024, pp. 420–26. ijsrcseit.com, https://doi.org/10.32628/CSEIT2410235.
31. Gong, Xulu, and Shujuan Zhang. “A High-Precision Detection Method of Apple Leaf Diseases Using Improved Faster R-CNN.” Agriculture, vol. 13, no. 2, Jan. 2023, p. 240. DOI.org (Crossref), https://doi.org/10.3390/agriculture13020240.
32. Pareek, Piyush Kumar, et al. “Clustering Based Segmentation with 1D-CNN Model for Grape Fruit Disease Detection.” 2023 IEEE International Conference on Integrated Circuits and Communication Systems (ICICACS), 2023, pp. 1–7. IEEE Xplore, https://doi.org/10.1109/ICICACS57338.2023.10099916.
33. Ahmed, Khandakar Rabbi, et al. “An Identification of Guava Fruit Disease Using ML.” 2024 3rd International Conference on Advancement in Electrical and Electronic Engineering (ICAEEE), 2024, pp. 1–6. IEEE Xplore, https://doi.org/10.1109/ICAEEE62219.2024.10561747.
34. Kumar, Varun, et al. “Hybrid Model for EffectivePapaya Leaf Disease Diagnosis.” 2024 International Conference on E-Mobility, Power Control and Smart Systems (ICEMPS), 2024, pp. 01–05. IEEE Xplore, https://doi.org/10.1109/ICEMPS60684.2024.10559298.
35. Gupta, Radhika, et al. “Lemon Diseases Detection and Classification Using Hybrid CNN-SVM Model.” 2023 Third International Conference on Secure Cyber Computing and Communication (ICSCCC), 2023, pp. 326–31. IEEE Xplore, https://doi.org/10.1109/ICSCCC58608.2023.10176828.