[1] Manju Mathur

[2] Rahul Jain

"A Comparative Analysis of Deep Learning Algorithms for Fruit Disease Classification"

Journal of Electrical Systems

Abstract: - Agriculture plays a pivotal role in the Indian economy, serving as the primary source of income for a substantial portion of the population. Enhancing fruit output is crucial for sustaining agricultural livelihoods. However, fruit diseases, mainly caused by fungal and bacterial pathogens, significantly impact fruit quality and overall yield. Timely identification of these diseases is essential for forecasting and mitigating their occurrence, leading to cost savings for farmers. Researchers have developed fruit disease identification systems to safeguard agricultural investments. This study aims to conduct a comprehensive comparative analysis of deep learning classification approaches for fruit disease detection. We evaluate the performance of VGG16, InceptionV3, MobileNetV2, ResNet50, NasNetV2, and a Convolutional Neural Network (CNN) model. Additionally, we incorporate optimization techniques such as Stochastic Gradient Descent with Momentum (SGDM), Adaptive Moment Estimation (Adam), and Root Mean Square Propagation (RMSProp), along with baseline learning techniques and transfer learning methods. In this study, we evaluate the performance of the ResNet50 architecture in fruit disease detection, achieving an impressive accuracy of 93.25%. This performance is compared to other deep learning architectures, including VGG16, InceptionV3, MobileNetV2, and NasNetV2. Our findings highlight the effectiveness of ResNet in accurately detecting fruit diseases, showcasing its potential as a valuable tool for farmers in mitigating the impact of diseases during the early stages of infestation.

Keywords: fruit diseases, deep learning, ResNet50, VGG16, InceptionV3, MobileNetV2, NasNetV2, optimization techniques.

I INTRODUCTION

Every country's economic growth and general advancement are greatly influenced by agriculture [1].Numerous substantial obstacles are continually encountered in the transportation of agricultural commodities throughout various global locations. Modern farmers may use better decision-making tools and advancements in automation to help them integrate their commodities, skills, and services more smoothly, which will increase their production, efficiency, and profitability [2].this statement sums up the idea of "smart farming" quite well. Improving fruit yield aims to increase total fruit production, which will help the agricultural sector progress. The agricultural industry plays a crucial role in driving growth in the economy, creating jobs, and facilitating trade between countries. Every sector of the economy relies on this function. sincerely want to increase fruit output on farmland, then must make sure that fruit farming is sustainable in terms of disease control [3, 4]. The amount of minerals and nutrients inside fruits has a considerable impact on the cultivation methods employed for apples, oranges, grapefruits, pomegranates, and plums. Magnesium, copper, phosphorus, calcium, potassium, and nitrogen are just a few of the essential minerals found in abundance in fruits. Many different kinds of fruit can contain the mineral iron. Damage to fruit growth, yield, and quality can result from essential nutrients not being readily available. There is an estimated 16.0% fruit loss due to the cold chain infrastructure in a year's time. It is critical to take precautions to quickly detect the start of the disease, so that the right actions may be put in place to protect the fruits from contamination. The illness may be properly contained and not spread to other fruit kinds or the world at large if this precaution was taken. A number of diseases, including as thread blight, rot, black spot, anthrax, and canker, can drastically lower fruit output. Other diseases that could be observed in this setting include rot and black spot. Harvesting, pruning, and spraying have not been automated using agricultural technology in the agricultural industry in recent years. For maximum fruit production, it is essential to promptly identify fruit illnesses and accurately predict when they will start [11, 16].

Modern refrigeration and shipping methods have made it possible to ship a broad range of fruits all across the world and this trend is just going to keep going. There is currently an urgent need to ensure the highest level of export efficiency, which is mostly achieved through the use of visual inspection carried out by skilled professionals. For the most part, people who are well-versed in the disease in question will visually inspect fruit without the aid of magnifying glasses in order to identify and classify it. Because of the farms' far-flung locations, the current undertaking is fraught with difficulty and expense.

Due to the isolated locations of these services, seeking advice from experts may be too time-consuming and expensive in many parts of the world's less developed nations. The high expense of travel is a potential cause of this occurrence. Identifying illnesses in fruit at their earliest stages requires automated detection systems that can monitor the fruit's development in real time [19]. The presence of illnesses inside the fruit could cause significant

[1,2] Department of Electronics &Communication Engineering, JECRC University, Jaipur

[1] manjumathur1108@gmail.com, [2] rj8134@gmail.com

losses, both in terms of quantity and quality, when it is harvested. An in-depth familiarity with the phenomena under study is necessary for the purpose of reducing any monetary losses in the coming year. A tree's leaves, branches, and trunk could all be infected if an illness spreads to nearby areas inside the tree [6]. A few of common diseases that can cause serious harm to apple orchards are apple scab, apple rot, and apple blotch and similar conditions. Apples can develop black spots of scale, which are commonly known as scabs [4]. Apple rot infections manifest as little depressions encircled by a red halo and shaped like round brown spots. You can spot these spots all throughout the apple. Fungi cause apple blotch, a disease that shows up as uneven or lobed borders on the fruit's exocarp, discolored flesh, or both [24, 25]. Scabies is a fungal infection that causes apple blotch, a disease affecting plants. Visual examinations of fruits are being carried out within the sector using computer vision. The automation of this technique allows for the evaluation of fruit color and size simultaneously. The constant presence of either a stem or a calyx, the wide diversity of fault kinds, and the intrinsic variances in the natural coloration of fruit skins among species make flaw detection a tough task. A fruit's overall condition and the presence or absence of any internal diseases should be carefully evaluated. Individuals can improve their capacity to control infections and increase the efficacy of therapeutic interventions by following proper management procedures, which may include the use of fungicides, insecticides, and other chemically-based treatments. To make plant disease prevention and management more effective, a variety of approaches can be applied (22, 23). Two well-known examples of the methods discussed before are imaging and spectroscopy.

Farmers are always looking for new ways to reduce labor needs while keeping production levels the same. This is due to the fact that the agricultural industry is becoming more and more commercialized. Because it has the ability to drastically cut down on the total cost of the activity, autonomous harvesting equipment should be seriously considered. The harvesting sector is the principal user of fruit identification technology, especially in the field where robotic systems are dominating. There are a number of potential applications for this method that share basic similarities with it, including illness diagnosis, maturity assessment, yield tracking in trees, and others [27].

Through the utilization of technical tools, farmers are able to access and collect data from reliable agricultural sources. As a result, they achieve outstanding success in their farming operations and reap large financial rewards. A more beneficial expansion in terms of financial gains may be possible with the application of these tactics. Precision agriculture has the potential to greatly improve our ability to detect plant diseases, fruit problems, and insect infestations. In addition, thanks to advancements and dissemination across various fields, it helps farmers access economically viable information and control strategies. This means it can effectively support farmers in delivering enough information and control methodologies at a low cost. Farmers suffer huge financial losses due to two main issues: insufficient fruit yield and the spread of fruit-related diseases. It is critical to diagnose fruit infections quickly and accurately in order to reduce the spread of disease and the amount of damage that fruits sustain [14]. Future generations of fruit growers will be able to produce and preserve more fruit because to technological advancements in their tools and techniques. Their financial resources will be better utilized thanks to this increased capability, leading to greater efficiency. In light of this, identifying the precise disease impacting the fruit is of paramount importance [2]. This in-depth essay analyzes the pros and cons of various procedures used to diagnose fruit illnesses and gives a detailed review of each. In addition, the author gives a detailed breakdown of the pros and cons of each method.

II LITERATURE SURVEY

In order to achieve this goal, **Khattak et al. [1]** employed a model they called Convolutional Neural Networks (CNNs). It is said that the CNN model's goal is to identify healthy citrus fruits and leaves and those with common citrus diseases like black spot, canker, blister, greening, or Melanose. Integrating many layers to extract complementing properties is key to the CNN architecture that has been developed. Both the Citrus and PlantVillage datasets were used to evaluate the CNN approach in comparison to other advanced DL methods. The research proves that the CNN Model is better than its competitors on multiple metrics. With an accuracy of 94% in testing, the CNN Model is a valuable decision-support tool for citrus farmers looking to identify diseases that harm the fruit or leaves.

Muhammad Zia Ur Rehman et al. (2) delivered a fresh, DL-based method for citrus disease classification. Two separate, previously trained deep learning algorithms were employed throughout our investigation. They apply picture enhancing techniques to broaden the citrus dataset that was previously used. Additional enhancements to the photographs' visual quality have been made using a hybrid stretching approach. Along with feature fusion allowing for the growth of the feature set, transfer learning allows for the retraining of pre-existing models. In order to make the models better, they apply both of these methods. When all of the features are considered together, the Whale Optimization approach (WOA), a meta-heuristic, is used to get the best possible performance. Citrus fruits are susceptible to six different illnesses, which can be classified according to one of the above characteristics. Outperforming other newly-established systems, the proposed method boasts a classification accuracy rating of 95.7.

An exhaustive review of the methods used to diagnose and classify illnesses affecting citrus tree foliage is presented in the academic publication by Saini, A.K. et al. (201X). Moreover, a comprehensive taxonomy of diseases that damage citrus fruit leaves is presented in this work. Automated diagnosis and classification of health problems is also a part of this research. Feature extraction, feature aggregation, classification, and preprocessing are some of the numerous methodologies that are now being explored in this study. A new method for identifying fruit diseases was suggested by Dubey et al. (2014) and later confirmed by experiments. The image processing method that has been suggested has multiple important steps. The image flaws are initially segmented using a K-Means clustering technique. The second step is to use the segmented image to extract cutting-edge features. Lastly, the photos are classified using a Multi-class support vector machine (SVM). In this apple disease laboratory case, we looked at and evaluated three different kinds of apple disease: scab, blotch, and rot. As a model, apple diseases were utilized in the investigation. The testing findings show that the method works well for automatic identification and accurate diagnosis of fruit diseases, thus it's a big aid in this field. Up to ninety-three of the objects that were queried can be accurately classified utilizing the presented methodology. Han, L. is the name of the person in issue. The researchers and their colleagues [5] came up with a novel approach to automatically detect agricultural illnesses using computer vision in a groundbreaking study. Markers control the watershed classification in this system, while superpixels regulate the feature evaluation and classification. According to the study's findings, the suggested method keeps processing speeds high while effectively detecting, evaluating, and determining the severity of crop illnesses.

Metin et al. [6] created a more effective version of their Faster R-CNN framework by adjusting the values of CNN parameters. Also included in the article is a Faster R-CNN architecture that aims to automate sugar beet leaf spot infection detection. Training and testing of the approach developed for the diagnosis of sickness severity utilizing imaging-based systems with specialized expertise involved 155 images in total. The results of the testing show that the method worked as expected, with a total accurate classification rate of 95.48% being very impressive. Additionally, the demonstrated method suggested that the Faster R-CNN framework may be enhanced by tailoring the CNN parameters according to the image properties and the particular regions that need detection. Having the method in place lends credence to this claim. When compared to the current approaches, the suggested approach produced better results for the important variables. So, it's reasonable to assume that this method could shorten the time needed to determine how bad the sugar beet leaf spot disease is in the main growing regions. In addition, the time and effort needed to assess the severity and course of the disease should be lessened with the help of the suggested methodologies.

Lorente et al. (7) examined the possibility of using visible and near infrared (NIR) reflectance spectroscopy to independently detect fungus Penicillium digitatum-induced deterioration in orange fruit. The reflectance spectra of sound and dissolving surface portions of mandarins cv are examined in this work. 'Clemenvilla' data was collected from two separate spectral regions: one covering the visible and near-infrared (NIR) wavelengths (650-1050 nm), and the other covering just the NIR range (1000-1700 nm). A few examples of spectral ranges are shown below. Comparing the sound and dissolving skin spectra revealed significant disparities in both sections of the spectrum. Three manifold learning methodsprincipal component analysis (PCA), factor analysis, and Sammon mapping were employed to transform high-dimensional spectrum data into visually appealing, lower-dimensional representations while preserving fundamental information. For this purpose, these tactics were employed. To determine whether skin is healthy or unhealthy, a supervised classifier that employs linear discriminant analysis uses feature vectors, which are low-dimensional representations of the data. Among the several classification methods used, NIR spectra analysis produced the best accurate results (97.8% accuracy). In addition, the specimens that were accurately categorized, regardless of their condition, reached percentages of 94 and 100, respectively. These results suggest that modern commercial citrus fruit sorting systems can sense rot using reflectance spectroscopy, which was previously unavailable.

Jahanbakhshi et al. (2018) identified sour lemon fruit faults, assessed the severity of these issues, and developed a more efficient system using an improved version of the Convolutional Neural Network (CNN) method. Images of sour lemons were initially separated into two groups: those showing healthy fruit and those showing unhealthy fruit in order to find any possible errors. The photos were subsequently classified using an improved Convolutional Neural Network (CNN) method after the initial processing stage. By incorporating data augmentation and stochastic pooling approaches, CNN effectively enhanced its performance. Feature extraction approaches and classification procedures such as k-nearest neighbor (KNN), artificial neural network (ANN), fuzzy, support vector machine (SVM), and decision tree (DT) were utilized to assess the supplied model in comparison to well-established methodologies. The CNN's credibility was confirmed to be 100 percent. Thus, image processing methods and CNNs (Convolutional Neural Networks) are crucial for improving sour lemon grading procedures and reducing waste.

Qimei Wang et al. [9] employed object recognition techniques and deep convolution neural networks (CNNs) for illness detection in tomatoes. Faster R-CNN is used for tomato infection identification, while Mask R-CNN is used for tomato affected area detection and segmentation. The best method for detecting tomato diseases has been determined by combining two item recognition algorithms with four deep Convolution Neural Networks (CNNs). It is common practice to separate data collected from the Internet into three groups while conducting investigations: training data, validation data, and test sets. In addition to differentiating between eleven different

tomato illnesses, the proposed methods may also identify the specific geographic areas and morphological traits linked to each disease.
In a study conducted by Nasir et al. (2010), Different kinds of fruits and diseases were classified using a Deep Neural Network (DNN) that made use of contour characteristics. A dataset consisting of photos of plants was used to fine-tune and pretrain the VGG19 deep learning classifier. Important features were successfully retrieved from the dataset using this method. After that, we used a serial-based technique to reconstruct the pyramid histogram of oriented gradient contour features, and then we integrated that data with the deep features. A definitive classification was made possible by first incorporating irrelevant qualities into the fusion process, and then using a "relevance-based" optimization method to extract the most relevant features from the combined vector. With a 99.6 percent success rate across multiple classification algorithms, the suggested method outperformed its predecessors.

2. MATERIALS AND MODELS

Datasets.
There are two kinds of fruit datasets that were employed in this study. For example, there is the publicly available FIDS-30 dataset [17], which uses images sourced from the Internet. The added noises in the 30–40 photographs that make up each category vary greatly. Ten different types of fruits, including apple, avocado, banana, cherry, grape, guava, mango, papaya, pineapple, and strawberry, are selected as the subjects for fruit disease detection using deep learning techniques. This selection allows for a diverse representation of common fruits found in agricultural settings, enabling comprehensive testing and analysis of the deep learning models' effectiveness in identifying diseases across various fruit types. By focusing on these fruits, the study aims to develop robust and accurate disease detection models that can benefit farmers by providing early detection and intervention strategies to safeguard their crops and optimize fruit production.

Table 1 number of dataset

Fruit	Quantity
Strawberry	400
Apple	500
Avocado	300
Cherry	700
Papaya	750
Mango	650
Banana	710
Grape	640
Pineapple	780

deep learning Model Descriptions
CNN Architecture- Neural network architectures include CNN. As DL technology controller, it is mostly effective in handling image organization challenges. One variant on the typical artificial neural network (ANN) is convolution neural networks (CNNs), which focus on pattern recognition in image modeling. Their main benefit over regular feed forward neural networks is the drastically reduced number of structural pieces (artificial neurons) needed thanks to their hierarchical approach. CNN, often known as a feed-forward algorithm, is an excellent method for detection. Both the network architecture and the training settings are simple and uncomplicated. One highly efficient improvement in detection is CNNs. Conversely, simplicity is achieved at the expense of complexity or a high number of weights. The five-layer internal architecture of a convolutional neural network (CNN) is shown here: an input layer, a folding layer with an establishing function, a pooling layer, a completely related layer, and a Softmax layer [24].
Layers:
- Input Layer: Accepts input images.

Convolutional Layers:
- Apply convolution operations to extract features from the input images.

Activation Functions:
- Introduce non-linearity into the model (e.g., ReLU).

Pooling Layers:
- Downsample feature maps to reduce computational complexity.

Fully Connected Layers:
- Perform classification based on the extracted features.

Softmax Layer:

- Generate probability distributions over the output classes.

Convolution: Hi(t)=Conv(X(t),Wi(t))+Bi(t)
Activation: Hi(t)=ReLU(Hi(t))
Pooling: Pi(t)=Pooling(Hi(t))
Fully Connected: Output (t)=Softmax(FC(Plast(t)))
VGG16- The VGG16 (Visual Geometry Group 16) architecture, proposed by K. Simonyan and A. Zisserman from the University of Oxford, is a deep convolutional neural network designed for image classification tasks. It is known for its simplicity and effectiveness. Below is a brief description of the VGG16 architecture:
Input Layer: Accepts input images of fixed size (e.g., 224x224 pixels).
Convolutional Layers (Conv): Consists of 13 convolutional layers, each followed by a rectified linear activation function (ReLU) for introducing non-linearity.
Max Pooling Layers: After every two convolutional layers, max pooling layers are applied to downsample the feature maps.
Fully Connected Layers (FC): The convolutional layers are followed by three fully connected layers for classification.
Output Layer: The output layer consists of softmax activation for multi-class classification, generating probability distributions over the output classes.
Layers:
Input Layer:
Accepts input images of fixed size (e.g., 224x224 pixels).
Convolutional Layers (Conv):
Consists of 13 convolutional layers, each followed by a rectified linear activation function (ReLU) for introducing non-linearity.
Max Pooling Layers:
After every two convolutional layers, max pooling layers are applied to downsample the feature maps.
Fully Connected Layers (FC):
The convolutional layers are followed by three fully connected layers for classification.
Output Layer:
The output layer consists of softmax activation for multi-class classification, generating probability distributions over the output classes.
Convolutional Operation (Conv): Hi(t)=Conv(,$X^{(t)}$,Wi(t))+ $B_i^{(t)}$
Activation (ReLU): $H_i^{(t)}$=ReLU$H_i^{(t)}$
Max Pooling: $P_i^{(t)}$=MaxPooling $H_i^{(t)}$
Fully Connected: Output $^{(t)}$ =Softmax (FC ($H_{last}^{(t)}$
ResNet-ResNet is not like other network topologies such as AlexNet, Over Feat, or VGG. ResNet is an example of a "exotic architecture" that is based on a micro-architecture model, also referred to as a "in-network architecture."
Inception - Input Layer: Accepts input images of variable size.
Inception Modules: The key component of the architecture is the inception module, which contains multiple parallel convolutional layers of different kernel sizes (1x1, 3x3, 5x5) and a max pooling layer. The outputs of these parallel layers are concatenated along the channel dimension.
Down sampling Layers: Between inception modules, down sampling layers (typically using 3x3 convolutions with strides) are applied to reduce spatial dimensions.
Fully Connected Layers (FC): The network ends with a global average pooling layer followed by fully connected layers for classification.
Output Layer: The output layer consists of softmax activation for multi-class classification.
Mathematical Functions:
Inception Module:

$$H_i^{(t)}=\text{Concatenate }(\text{Conv1x1}(X(t)),\text{Conv3x3}(X(t)),\text{Conv5x5}(X(t)),\text{MaxPooling}(X(t)))$$

Down sampling:
$P_i^{(t)}$Conv3x3_Stride$H_i^{(t)}$
Global Average Pooling:
$P_{iast}^{(t)}$= GlobalAveragePooling$P_i^{(t)}$
Fully Connected:
Output (t)=Softmax(FC$P_i^{(t)}$)

MobileNet- When it comes to embedded vision and mobile applications, nothing beats MobileNet, a convolutional neural network architecture. Howard et al. had the idea to introduce it. To decrease computing cost and model size while keeping acceptable accuracy, MobileNet uses depth wise separable convolutions. This is the core idea underlying the network. The MobileNet architecture is detailed below:

Layers:

Depthwise Separable Convolution Block:

Hi=Pointwise (Depthwise (Xi))

Where Xi is the input feature map, Depthwise represents the depthwise convolution, and Pointwise represents the pointwise convolution.

Downsampling:

Downsampling layers can be achieved using strided convolutions or other techniques to reduce spatial dimensions.

Global Average Pooling:

Plast=GlobalAveragePooling (Hi)

Fully Connected:

Output=Softmax(FC(Plast))

NASNet-In NASNet, although the general structure is defined as shown above, the barriers or elements are not defined by the author in advance. Instead, they are searched through a series of affirmative research methods. The number of phantom repeats N and the number of initial convolution filters are used as free parameters for scaling. Particularly, these cells are called regular cells or degenerative cells. Repeat the convolution element on a feature map of the same size. Returns to the convolution unit of the feature map, which reduces the height and width of the event map by twice. The regulatory RNN (repetitive neural network) searches only for the structure (or interior) of normal and degenerative cells.

Key Features:

Automated Architecture Search: NASNet employs reinforcement learning or evolutionary algorithms to automatically search for optimal network architectures.

Cell-based Architecture: NASNet architecture is composed of repeating cells, each representing a building block of the network. These cells are stacked together to form the complete network architecture.

Flexible and Scalable: NASNet allows for flexibility in architecture design and can adapt to different computational constraints and performance requirements.

High Performance: NASNet architectures have demonstrated superior performance on image classification benchmarks like ImageNet.

Architecture Components:

Normal Cell: A normal cell is a basic building block of the network, containing a sequence of convolutional and pooling operations.

Reduction Cell: Similar to the normal cell, but typically includes operations to reduce spatial dimensions (e.g., strided convolutions) to decrease computational cost.

Skip Connections: Skip connections or residual connections may be included within cells to facilitate gradient flow and improve training stability.

Weight Sharing: NASNet often utilizes weight sharing mechanisms to reuse learned parameters across different parts of the architecture, reducing the number of trainable parameters.

Cell Architecture: Each cell is represented by a directed acyclic graph (DAG), where nodes represent operations (e.g., convolution, pooling) and edges represent data flow between operations. The architecture search process aims to optimize the connectivity pattern and operation choices within each cell to maximize performance.

Search Algorithm:NASNet employs reinforcement learning or evolutionary algorithms to search the space of possible architectures. The search algorithm evaluates different candidate architectures based on their performance on a validation set and updates the search policy to guide the search towards more promising architectures. After the search process, the best-performing architecture is selected based on validation performance and deployed for inference.

IV TRAINING, OPTIMIZER AND LEARNING METHOD

In fruit disease detection, training involves the process of feeding labeled datasets into a convolutional neural network (CNN) architecture, where the network learns to extract relevant features from input images. Optimization techniques such as Stochastic Gradient Descent with Momentum (SGDM), Adaptive Moment Estimation (Adam), and Root Mean Square Propagation (RMSProp) are commonly employed to adjust the network parameters iteratively, minimizing the loss function and enhancing model performance. Baseline learning techniques, typically involving traditional machine learning algorithms or simpler neural network architectures, serve as comparators to evaluate the effectiveness of more complex models like CNNs. Additionally, transfer learning, a method where a pre-trained model's knowledge on a large dataset is transferred to a related task, can be leveraged to improve the performance of fruit disease detection models, especially when labeled training data is limited. By integrating these training, optimization, and learning methods, fruit disease detection systems can achieve higher accuracy and robustness in identifying and classifying diseases, thereby aiding farmers in timely and accurate disease management decisions.

Baseline Training: In this technique, the pre-trained models are trained from without leveraging pre-trained weights and other tasks. This permit for an assessment of the models' performance without any aforementioned domain-specific information. indicate the parameters of the NLP model as θ. The baseline training involves minimizing the following loss function J(θ) during gradient descent: [37]

$$\theta_t+1=\theta_t-\alpha\nabla J(\theta_t)$$

Where α is the learning rate, and ∇J(θt) is the gradient of the loss with respect to the parameters.

Transfer Learning [TL]:TL is process of by before learned models on a divide but related task and then fine-tuning those models such that they are suitable for the depression classification problem at hand. This approach makes use of the information that was acquired during the first training, which has the potential to improve the models' aptitude for comprehending and categorizing material that is linked with depression. TL is process of adjusting a model that has already been trained to perform a new task. The overall objective is to minimize a combined loss function Jtotal(θ), which is a sum of the pre-trained model's loss pre-trained Jpre-trained(θ) and the task-specific loss task-specific(Jtask-specific(θ):

$$\lambda J_{\mathrm{pre-trained}}(\theta)+(1-\lambda)\, J_{\mathrm{task-specific}}(\theta)$$

Where:

Jtotal(θ) is the combined loss function.

Jpre−trained(θ) is the loss of the pre-trained model.

Jtask−specific(θ) is the task-specific loss.

λ is a hyperparameter controlling the balance between the two losses.

Where λ is a hyperparameter controlling the balance between the two losses.

Optimization Algorithms:

Stochastic Gradient Descent with Momentum (SGDM): SGDM is an optimization algorithm that combines the advantages of SGD with a momentum term. The momentum helps pick up the pace convergence by accumulating gradients from previous steps, enabling faster movement through the parameter space.

$v_{t+1} = \beta_{vt}+(1-\beta)\,\nabla J\,\theta_t$

$\theta_{t+1} = \theta_t\text{-}av_t+1$

Where β is the momentum term, α is learning rate, ∇J(θt) is the gradient of the loss, and vt is the momentum term.

Adaptive Moment Estimation (ADAM): ADAM is an approach for adaptive optimization that modifies the learning rates for each parameter using a separate algorithm. Adaptive learning rates are provided, and quicker convergence is often achieved, thanks to the amalgamation of concepts from momentum and RMSprop.

$\mathbf{m_{t+1} = \beta_1 m_t+(1-\beta_1)\,\nabla J(\theta_t)}$

$\mathbf{v_{t+1} = \beta_1 v_t+(1-\beta_1)\,\nabla J(\theta_t)}$

$\mathbf{\hat{m}_{t+1} = \frac{1-\beta_1^{t+1}}{1-\beta 1} m_{t+1}}$

$\mathbf{\hat{v}_{t+1} = \frac{1-\beta_1^{t+1}}{1-\beta 1} v_{t+1}}$

β1and β2 are exponential decay rates.

∇J(θt) is the gradient of the loss.

mt and vt are the first and second moment estimates respectively.

$\mathbf{\hat{m}_{t+1}}$, $\mathbf{\hat{v}_{t+1}}$ are bias-corrected moment estimates.

Root Mean Square Propagation (RMSprop): RMS prop is an adaptive learning rate optimization algorithm. It maintains a moving average of squared gradients for each parameter. This helps standardize the learning rates based on the past gradients, preventing the learning rates from fetching too large.

$v_{t+1} = \beta_{vt}+(1-\beta)\,\nabla J\,\theta_t^2)$

θt+1=θt−αvt+1+ϵ∇J(θt)

Where β is an exponential decay rate, α is the learning rate, ϵ is a small constant.[32]

RESEARCH METHODOLOGY

The proposed system for fruit image analysis integrates various preprocessing techniques, including input image preprocessing and Histogram of Oriented Gradients (HOG) feature extraction, to enhance the quality and relevance of input data. Subsequently, segmentation and classification processes are employed, leveraging optimization techniques such as Stochastic Gradient Descent with Momentum (SGDM) and Adaptive Moment Estimation (Adam), along with baseline learning and transfer learning methods. These techniques enable the system to accurately segment fruit images and classify them based on disease. By harnessing the power of deep learning models trained on large datasets, the proposed system aims to achieve high accuracy and robust performance in fruit image analysis, facilitating efficient disease detection and quality assessment for agricultural applications.

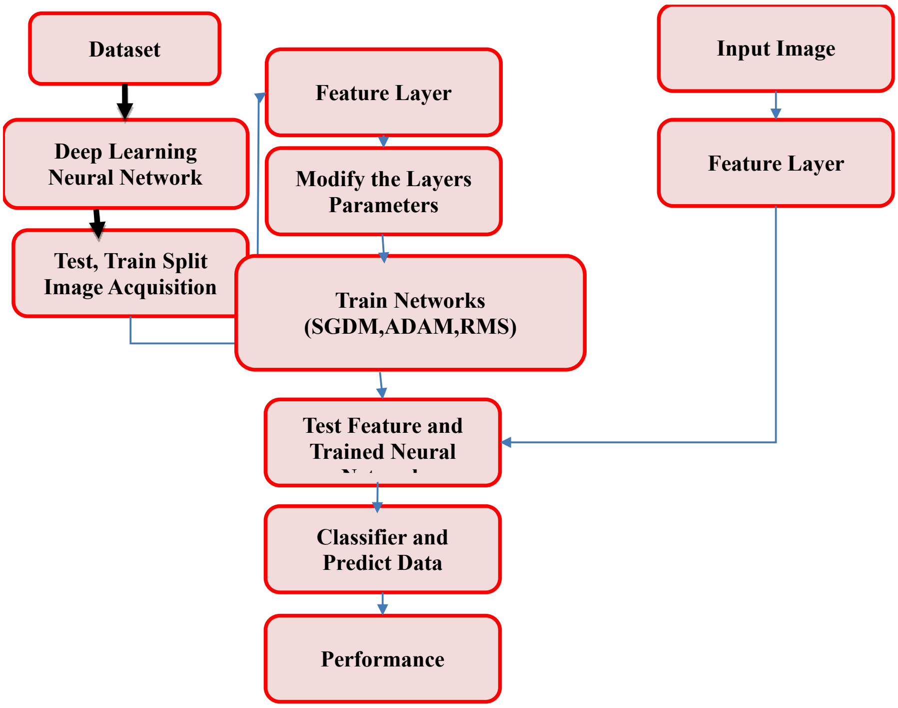
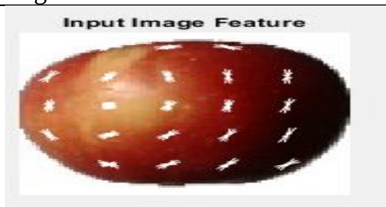
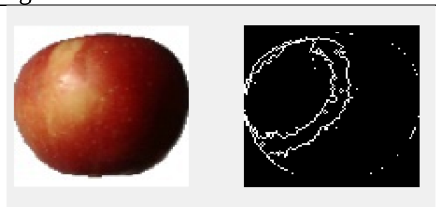
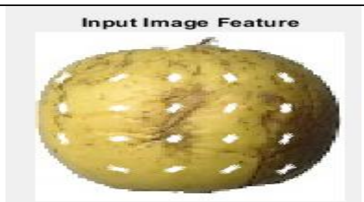
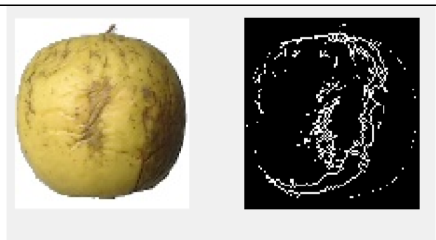
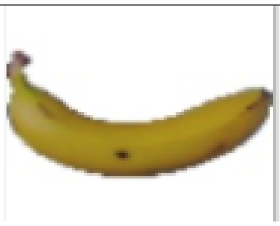
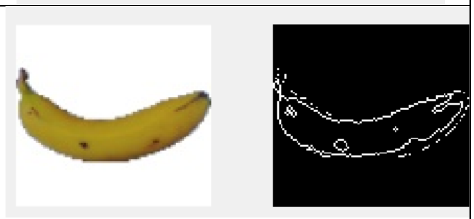
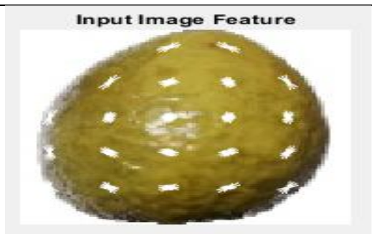


Table 2 pre-processing and classification results

Input image	Hog feature	Segmentation
		
		
		
		

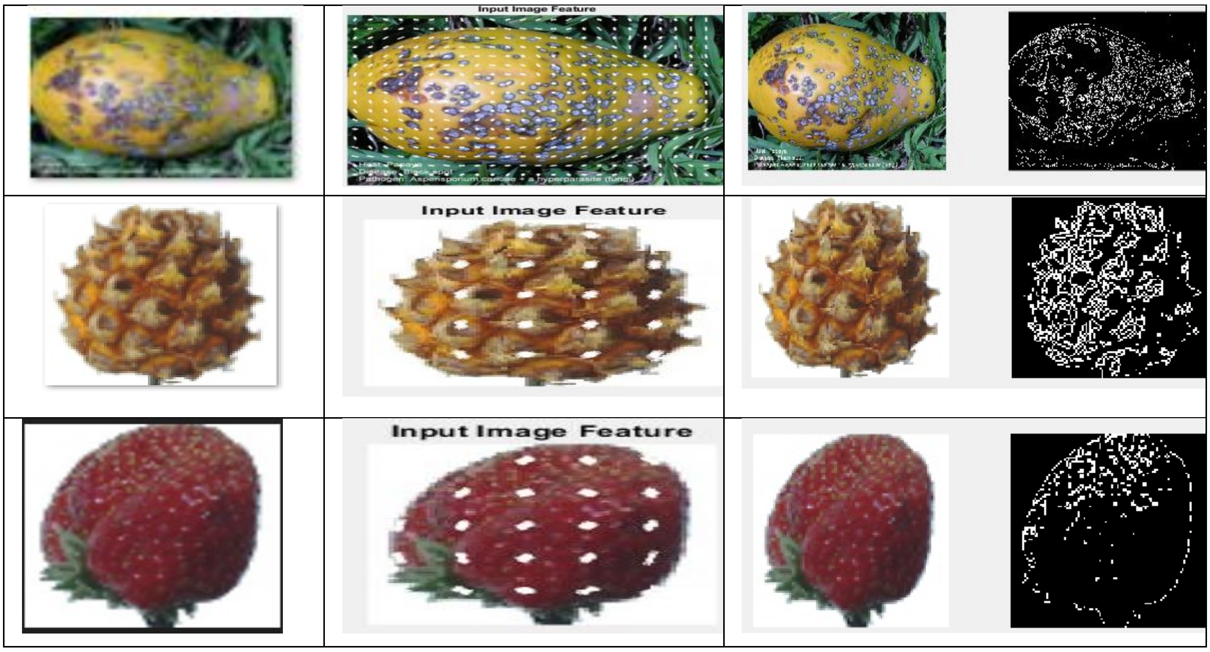


Table 3 Pre-Trains Models Performance

Deep learning Architecture	Learning Method	Training Method	Validation Accuracy
VGG 16	Baseline	ADAM	80%
		SGDM	90%
		RMS Propagation	77%
	Transfer Learning	ADAM	85%
		SGDM	92%
		RMS Propagation	90%
InceptionV3	Baseline	ADAM	91%
		SGDM	85%
		RMS Propagation	90%
	Transfer Learning	ADAM	92%
		SGDM	93%
		RMS Propagation	91%
MobileNetV2	Baseline	ADAM	93%
		SGDM	92%
		RMS Propagation	90%
	Transfer Learning	ADAM	90%
		SGDM	91%
		RMS Propagation	91%
ResNet50	Baseline	ADAM	93%
		SGDM	92%
		RMS Propagation	90%
	Transfer Learning	ADAM	94%
		SGDM	91%
		RMS Propagation	94%

NasNetV2	Baseline	ADAM	93%
		SGDM	90%
		RMS Propagation	92%
	Transfer Learning	ADAM	92%
		SGDM	92%
		RMS Propagation	91%
CNN	Baseline	ADAM	93%
		SGDM	91%
		RMS Propagation	86%
	Transfer Learning	ADAM	91%
		SGDM	86%
		RMS Propagation	93%

The table presents the performance of pre-trained models across different deep learning architectures, including VGG16, InceptionV3, MobileNetV2, ResNet50, NasNetV2, and a Convolutional Neural Network (CNN). For each architecture, the table outlines the learning method (ADAM, SGDM, RMS Propagation) and the corresponding training method (Baseline or Transfer Learning), along with the validation accuracy achieved.

Table 3 classification and segmentation performance for SGDM Training and transfer Techniques

Pre-Trained Model	Sensitivity	Specificity	Precision	Recall	Jaccrad Coefficient	Dice Coefficient
VGG 16	90.33%	91.22%	90.88%	91.92%	91.02%	88.23%
InceptionV3	92.17%	90.88%	91.24%	89.26%	90.28%	92.29%
MobileNetV2	91.77%	88.64%	89.15%	92.65%	85.89%	91.56%
ResNet50	94.56%	94.66%	94.62%	93.92%	95.87%	94.77%
NasNetV2	91.35%	88.92%	91.24%	90.34%	90.75%	91.47%
CNN	87.98%	90.99%	92.98%	89.36%	89.66%	91.34%

Table 4 showing the classification and segmentation performance for ADAM Training Method and transfer Techniques

Pre-Trained Model	Sensitivity	Specificity	Precision	Recall	Jaccrad Coefficient	Dice Coefficient
InceptionV3	91.32%	92.55%	91.34%	90.10%	91.98%	92.57%
VGG16	91.70%	87.93%	89.33%	89.65%	91.87%	92.92%
ResNet50	95.95%	95.87%	94.55%	94.88%	95.77%	94.89%
DenseNet	82.88%	88.55%	89.24%	87.87%	91.21%	89.28%
GoogleNet	90.26%	91.11%	91.26%	92.38%	92.97%	89.28%

MobileNetv2	92.35%	92.55%	93.89%	91.25%	92.65%	91.98%

Table 5 showing the classification and segmentation performance for RMS Propagation Training Method and transfer Techniques

Pre-Trained Model	**Sensitivity**	**Specificity**	**Precision**	**Recall**	**Jaccard Coefficient**	**Dice Coefficient**
InceptionV3	94.25%	95.90%	93.71%	98.31%	95.54%	95.20%
VGG16	91.15%	80.89%	95.23%	86.40%	90.14%	91.33%
ResNet50	95.85%	96.12%	87.05%	95.23%	94.12%	96.50%
DenseNet	85.78%	94.78%	90.98%	84.45%	90.33%	90.32%
GoogleNet	94.60%	95.12%	92.12%	94.13%	94.43%	92.45%
MobileNetv2	91.85%	94.12%	87305%	84.23%	84.12%	92.50%

Table 6 classification and segmentation performance for SGDM Training Techniques

Pre-Trained Model	**Sensitivity**	**Specificity**	**Precision**	**Recall**	**Jaccrad Coefficient**	**Dice Coefficient**
InceptionV3	88.32%	90.24%	91.28%	91.92%	91.02%	88.23%
VGG16	90.65%	91.82%	91.73%	89.67%	90.78%	92.68%
ResNet50	93.67%	90.83%	94.73%	93.85%	95.88%	94.71%
DenseNet	91.71%	87.69%	87.76%	92.65%	91.80%	87.69%
GoogleNet	88.35%	89.92%	90.24%	91.34%	92.75%	90.78%
MobileNetv2	90.82%	91.88%	92.87%	88.83%	90.87%	91.83%

Table 7 showing the classification and segmentation performance for ADAM Training Method

Pre-Trained Model	**Sensitivity**	**Specificity**	**Precision**	**Recall**	**Jaccrad Coefficient**	**Dice Coefficient**
InceptionV3	90.68%	91.73%	92.76%	91.53%	88.98%	91.75%
VGG16	87.77%	86.93%	88.92%	88.82%	90.87%	88.83%
ResNet50	91.94%	91.87%	93.68%	34.81%	92.79%	90.80%

DenseNet	87.83%	88.78%	88.77%	96.84%	87.59%	89.85%
GoogleNet	91.71%	90.68%	92.83%	90.78%	89.80%	90. 84%
MobileNetv2	88.93%	91.79%	90.89%	92.83%	90.69%	92.80%

Table 8 showing the classification and segmentation performance for RMS Propagation Training Method

Pre-Trained Model	Sensitivity	Specificity	Precision	Recall	Jaccard Coefficient	Dice Coefficient
InceptionV3	92.89%	93.88%	92.71%	88.86%	87.66%	90.77%
VGG16	90.73%	88.78%	93.67%	87.80%	90.65%	91.67%
ResNet50	94.84%	91.86%	87.78%	90.70%	89.79%	91.69%
DenseNet	87. 87%	93.71%	90.82%	86.83%	90.83%	89.57%
GoogleNet	93.69%	89.86%	92.93%	93.77%	91.93%	80.86%
MobileNetv2	90.84%	91.94%	87.68%	89.83%	87.70%	91.89%

RESULT ANALYSIS

In the baseline training approach, where models are trained from scratch, each architecture demonstrates varying degrees of accuracy across different learning methods. Notably, VGG16 and ResNet50 exhibit relatively high accuracies, especially with the SGDM optimization technique.

In the transfer learning approach, where pre-trained models are fine-tuned on the specific task of fruit disease detection, the architectures generally achieve higher accuracy compared to baseline learning. ResNet50 stands out as the top-performing architecture in transfer learning, consistently achieving accuracies above 90% across different optimization techniques.

References

[1] Asad Khattak, Muhammad Usama Asghar, Ulfat Batool, Muhammad Zubair Asghar, Hayat Ullah, Mabrook Al-Rakhami, (Member, Ieee), And Abdu Gumae, Automatic Detection of Citrus Fruit and Leaves Diseases Using Deep Neural Network Model, IEEE access, VOLUME 9, 2021

[2] Muhammad Zia Ur Rehman, Fawad Ahmed, Muhammad Attique Khan, Usman Tariq, Sajjad Shaukat Jamal, Jawad Ahmad, and Iqtadar Hussain, Classification of Citrus Fruit Diseases Using Deep Transfer Learning, Tech sciences press, Computers, Materials & Continua, DOI:10.32604/cmc.2022.019046

[3] Ashok Kumar Saini, Roheet Bhatnagar, Devesh Kumar Srivastava, Detection and Classification Techniques of Pomegranate Leaves Diseases: A Survey, Turkish Journal of Computer and Mathematics Education, 3499 Vol.12 No.6 (2021), 3499-3510

[4] Shiv Ram Dubey ,Adapted Approach for Fruit Disease Identification using Images Han, L.; Haleem, M.S.; Taylor, M. A Novel Computer Vision-based Approach to Automatic Detection and Severity Assessment of Crop Diseases. 2015, on 20 December 2021).

[5] Metin, M.; Adem, K. Automatic detection and classification of leaf spot disease in sugar beet using deep learning algorithms.Physica A 2019, 535, 122537

[6] Lorente, D.; Escandell-Montero, P.; Cubero, S.; Gómez-Sanchis, J.; Blasco, J. Visible-NIR reflectance spectroscopy and manifold learning methods applied to the detection of fungal infections on pomegranate fruit. J. Food Eng. 2015, 163, 17–24.

[7] Jahanbakhshi, A.; Momeny, M.; Mahmoudi, M.; Zhang, Y.D. Classification of sour lemons based on apparent defects using stochastic pooling mechanism in deep convolutional neural networks. Sci. Hortic. 2020, 263, 109133.

[8] Qimei Wang, Feng Qi, Minghe Sun, Jianhua Qu ,and Jie Xue3Identification of Tomato Disease Types and Detection of Infected Areas Based on Deep Convolutional Neural Networks and Object Detection Techniques, Hindawi Computational Intelligence and Neuroscience Volume 2019, Article ID 9142753, 15 pages, https://doi.org/10.1155/2019/9142753

[9] Inzamam Mashood Nasir, Asima Bibi, Jamal Hussain Shah, Deep Learning-Based Classification of Fruit Diseases: An Application for Precision Agriculture,Tech science press, Computers, Materials &Continua DOI:10.32604/cmc.2020.012945

[10] Jamil Ahmad, Bilal Jan, Haleem Farman, Wakeel Ahmad and Atta Ullah, Disease Detection in Plum Using Convolutional Neural Network under True Field Conditions, MDPI 2020.

[11] Almetwally M. Mostafa, Swarn Avinash Kumar, Guava Disease Detection Using Deep Convolutional Neural Networks: A Case Study of Guava Fruits, MDPI 2021.

[12] Williams, H.A.; Jones, M.H.; Nejati, M.; Seabright, M.J.; Bell, J.; Penhall, N.D.; Barnett, J.J.; Duke,M.D.; Scarfe, A.J.; Ahn, H.S.; et al. Robotic kiwifruit harvesting using machine vision, convolutional neural networks, and robotic arms. Biosyst. Eng. 2019, 181, 140–156.

[13] Santos, T.T.; de Souza, L.L.; dos Santos, A.A.; Avila, S. Grape detection, segmentation, and tracking using deep neural networks and three- dimensional association. Comput. Electron. Agric. 2020, 170, 105247.

[14] Yu, Y.; Zhang, K.; Yang, L.; Zhang, D. Fruit detection for strawberry harvesting robot in non- structural environment based on Mask-RCNN. Comput. Electron. Agric. 2019, 163, 104846.

[15] Ganesh, P.; Volle, K.; Burks, T.F.; Mehta, S.S. Deep Orange: Mask R-CNN based Orange Detection and Segmentation; 6th IFAC Conference on Sensing, Control and Automation Technologies for Agriculture Agricontrol 2019. IFAC- Papers On Line 2019, 52, 70–75.

[16] Liu, Z.; Wu, J.; Fu, L.; Majeed, Y.; Feng, Y.; Li, R.; Cui, Y. Improved kiwifruit detection using pre- trained VGG16 with RGB and NIR information fusion. IEEE Access 2019, 8, 2327–2336.

[17] Ge, Y.; Xiong, Y.; From, P.J. Instance Segmentation and Localization of Strawberries in Farm Conditions for Automatic Fruit Harvesting; 6th IFAC Conference on Sensing, Control and Automation Technologies for Agriculture AGRICONTROL 2019. IFAC-Papers On Line 2019, 52, 294–299.

[18] Altaheri, H.; Alsulaiman, M.; Muhammad, G. Date fruit classification for robotic harvesting in a natural environment using deep learning. IEEE Access 2019, 7, 117115–117133.

[19] Zapotezny-Anderson, P.; Lehnert, C. towards Active Robotic Vision in Agriculture: A Deep Learning Approach to Visual Servoing in Occluded and Unstructured Protected Cropping Environments. IFAC-PapersOnLine 2019, 52, 120–125.

[20] Lin, G.; Tang, Y.; Zou, X.; Xiong, J.; Li, J. Guava detection and pose estimation using a low-cost RGB-D sensor in the field. Sensors 2019, 19, 428.

[21] Orchi H, Sadik M, Khaldoun M. On using artificial intelligence and the internet of things for crop disease detection: A contemporary survey. Agriculture. 2021 Dec 22;12(1):9.

[22] Abbas, A.; Jain, S.; Gour, M.; Vankudothu, S. Tomato plant disease detection using transfer learning with C-GAN synthetic images. Comput. Electron. Agric. 2021, 187, 106279.

[23] H. Wang, Q. Mou, Y. Yue and H. Zhao, "Research on Detection Technology of Various Fruit Disease Spots Based on Mask R- CNN," 2020 IEEE International Conference on Mechatronics and Automation (ICMA), 2020, pp. 1083-1087,doi: 10.1109/ICMA49215.2020.9233575.

[24] S. R. N. M. Ayyub and A. Manjramkar, "Fruit Disease Classification and Identification using Image Processing," 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC), 2019, pp. 754-758, doi: 10.1109/ICCMC.2019.8819789.

[25] N. Saranya, L. Pavithra, N. Kanthimathi, B. Ragavi and P. Sandhiyadevi, "Detection of Banana Leaf and Fruit Diseases Using Neural Networks," 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA), 2020, pp. 493-499,

[26] S. M. Jaisakthi, P. Mirunalini, D. Thenmozhi and Vatsala, "Grape Leaf Disease Identification using Machine Learning Techniques," 2019 International Conference on Computational Intelligence in Data Science (ICCIDS), 2019, pp. 1-6, doi: 10.1109/ICCIDS.2019.8862084.

[27] R. Ramya, P. Kumar, K. Sivanandam and M. Babykala, "Detection and Classification of Fruit Diseases Using Image Processing & Cloud Computing," 2020 International Conference on Computer Communication and Informatics (ICCCI), 2020, pp. 1-6, doi: 10.1109/ICCCI48352.2020.9104139

[28] Romi Morzelona. (2017). Evaluation and Examination of Aperture Oriented Antennas. International Journal of New Practices in Management and Engineering, 6(01), 01 - 07.

[29] Ramana, K. V. ., Muralidhar, A. ., Balusa, B. C. ., Bhavsingh, M., & Majeti, S. . (2023). An Approach for Mining Top-k High Utility Item Sets (HUI).International Journal on Recent and Innovation Trends in Computing and Communication, 11(2s), 198–203.https://doi.org/10.17762/ijritcc.v11i2s.6045

[30] Dhabliya, D., Soundararajan, R., Selvarasu, P., Balasubramaniam, M.S., Rajawat, A.S., Goyal, S. B., Raboaca, M. S., Mihaltan, T. C., Verma, C., Suciu, G. Energy-Efficient Network Protocols and Resilient Data Transmission Schemes for Wireless Sensor Networks—An Experimental Survey (2022) Energies, 15 (23), art. no. 8883.