International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

xACO_1DCNN: Ant Colony Optimization Based 1D-Convolution Neural Network based Fruit grading and Classification

[1]Vigneswara Reddy K, [2]Prof. Dr. A. Suhasini, [3]Dr.V.V. S. S. S. Balaram

Submitted: 26/05/2023 **Revised**: 08/07/2023 **Accepted**: 27/07/2023

Abstract: The automation of fruit classification represents a compelling application of computer vision. Historically, fruit classification methods have frequently depended on manual procedures that rely on human visual perception. However, these methods are characterized by their tedious nature, lengthy time requirements, and lack of consistency.The primary criterion for classifying fruits is their external shape and appearance.In recent years, the fruit industry has witnessed a growing utilization of computer machine vision and image processing techniques. These technologies have proven to be particularly valuable for various applications within the industry, such as quality inspection and the sorting of fruits based on colour, size, and shape. Research conducted in this field suggest that the utilisation of machine vision systems holds promise in enhancing product quality and alleviating the need for manual fruit sorting processes.This paper presents the design and development of an intelligent automation system for the grading of mango fruit.The fruit segmentation process is initially conducted using the U-NET model, which incorporates feature extraction from EffecientNet+ResNet101. The utilisation of the Artificial Bee Colony Optimisation algorithm has been employed in order tominimise and optimise the feature vector.The categorization of fruit quality has been accomplished by considering surface defects and maturity classification. Therefore, in order to classify defects, the 1D-CNN has been trained using optimal segmented features of abnormalities. To enhance the performance, the proposed ACO_1DCNN model is employed to optimise the classification accuracy. The proposed classifier is utilised to categorise the maturity levels as ripe, partially ripe, and unripe.Ultimately, the defect and maturity output have been employed as determinants for categorising the quality as either good, average, or bad. The experimental results demonstrate that the ACO_1DCNN model attains an accuracy of 97.95%, precision of 95.97%, recall of 95.59%, F1-score of 95.76%, AUC score of 99.42%, and MCC of 94.34% when evaluated on the testing dataset.During the training process, the model attains an accuracy of 97.62%, precision of 95.83%, recall of 94.96%, F1-score of 95.33%, AUC score of 99.56%, and MCC of 93.69%.

Keywords*- fruit classification, mango, feature extraction, ACO, Convolution Neural Network, Segmentation, deep learning.*

1. Introduction

- Due to their high vitamin and dietary fiber content, fruits are considered an integral part of the human diet [1]. There are more than 2,000 different types of fruits in the world, yet the average person only knows around 10% of them. According to data collected from throughout the world, in 2021, millions of metric tons of fruit were grown and gathered. The top three countries in terms of fruit output are China, India, and Brazil [2]. Farmers and the general public might benefit greatly from the introduction of a sophisticated fruit recognition system that uses a simple camera or sensor [3]. Classification and identification systems for fruits have gained popularity in the modern period, characterized by great technical advances, as useful instruments for educational participation among youngsters, generating tremendous attention [4]. Recent improvements in object detection and semantic image segmentation are made possible by employing deep neural networks in computer vision technologies [5]. In the field of image identification and classification, DNNs are used to identify and categorize photographs of fruits. When compared to other machine learning algorithms, DNNs perform exceptionally well. As a type of deep learning technique, Convolutional Neural Networks (CNNs) are becoming increasingly popular. It is generally agreed that Convolutional Neural Networks (CNNs) are the most common type of ANNs used in the deep learning sector.Video and picture recognition [7], facial recognition [8], handwritten digit recognition [9,10], and even fruit recognition [11] are just some of the visual recognition methods now in use. Several areas, such fruit recognition withCNN, have reached a degree of accuracy on par using human ability. The structure of CNNs is based on the biological model of mammalian visual systems.Cats' visual cortex cells, as identified by D. H. Hubel et al. in 1962, are extremely specialized, with each cell only reacting to stimuli in a very

1Ph.D.Research Scholar, Department of CSE, Annamalai University.
2Professor, Department of CSE, Annamalai University.
3Professor, Department of CSE, Anurag University.
Email:1vignesh2friend@gmail.com,2suha_babu@yahoo.com,3vbalaram23@gmail.com.

limited area of the visual field (the receptive field) [12].A pattern recognition model, the neocognitron was first proposed by Fukushima in 1980 [13]; it was influenced by the work of Hung et al. [14]. In order to extract and combine high-level characteristics from 2D input, the CNN is characterized by a network topology that includes convolution and pooling layers. Multiple neurons are included within each layer of a CNN, much like an ANN.A neuron in the next layer receives as input the weighted total of the outputs of all the neurons in the previous layer plus a bias factor.According to [15], the CNN layer has three distinguishing features. Because they are only connected to other neurons in their immediate vicinity, these neurons are displaying a pattern of partial connectivity. To train the network, a cost function is developed. What comes out of the network is contrasted to what should come out. Robotic harvesting and yield mapping are two areas where accurate fruit identification is of paramount importance [16]. Accuracy, trainability on a large and easily available dataset, real-time prediction, and flexibility are all need for a robust fruit identification system.Therefore, we have used a Convolutional Neural Network (CNN) to create a fruit identification classifier within the scope of our research. The RGB color channels make up the input image, which is taken at a resolution of 100x100 pixels. We tested the networks with five different sets of hidden layer configurations and compared their performance in terms of accuracy. The experiment's definitive results show a considerable increase in the participants' ability to identify fruits. Python and the TensorFlow library are used to construct the mathematical model of the network. In this section, we've outlined the main results of this study:

- To develop UNET based segmentation architecture for mango fruit identification from other fruits, in midst of colour confusion

- To grade according to its quality by using the color, shape and size features .Hence to identify the class of the fruit, ensemble feature extractor such as (EfficientNet, ResNet101) is used with feature fusion process

- The categorization of fruit quality can be achieved through the utilisation of computer vision and machine learning technology.The ID-CNN employed in this research serves the purpose of categorising various stages, namely good, average, and poor, which are determined by factors such as fruit maturity, texture, and geometric features.

2. Related Works

A SPP-Net and R-CNN-based segmentation algorithm were used together in a study [17] to classify mango quality. Convolutional Neural Networks (CNNs) and Fuzzy Classifiers were used together in a hybridized method in prior studies [18]. The hope behind this strategy was that it would lead to more precise mango categorization. Additionally, glowworm swarm optimization, which is self-adaptive, is used to choose features. Results from evaluating the proposed model on a variety of good and bad mango photos show that the proposed strategy is more accurate than competing methods across a number of important measures. Mango invariance is easily detectable using the MangoNet architecture [19]. The suggested approach in [20] uses a number of features to correctly determine the mango variety and place them into quality categories.Mango ripening stage is extracted using a Convolutional Neural Network (CNN). The collected characteristics are sent into a Random Forest classifier, which in turn uses them to determine whether or not a mango is of poor, average, or high grade. In [21], the authors present a deep learning approach to automate the categorization and grading of eight distinct mango cultivars. Color, size, form, and texture are all used as quality criteria in this method.Data augmentation methods such as picture rotation, translation, zooming, shearing, and horizontal flip were used in this investigation. The purpose of this study was to evaluate and contrast three popular 3-layer CNN architectures: VGG16, ResNet152, and Inception v3. To gauge these structures' efficacy, an evaluation was conducted using enhanced data. The goal of the Enhanced Fuzzy-based K-means clustering method used in this work [22] was to improve segmentation efficiency.Maximally Correlated Principal Component Analysis (MCPCA) is used for the feature selection procedure. Finally, the Backpropagation Based Discriminant Classifier (BBDC) is used to evaluate the most severely damaged regions. In order to reduce the size of detection networks, a generalized attribution technique is presented in [23]. Mango detection in real time is made possible by the suggested method's flexibility and ability to be fine-tuned.The original YOLOv3-tiny network's convolutional kernels closely correlated with certain target outputs can be determined by forming channel and spatial masks to broaden the scope of the attribution approach.

The techniques mentioned above may exhibit one or more of the following four limitations. Firstly, additional computational resources are required. Furthermore, it should be noted that the classifier may not be applicable to all types of fruits. Secondly, its capabilities are limited to identifying and distinguishing between different varieties within a given category. Thirdly, the recognition systems lack robustness due to the potential for different fruit images to possess similar or identical colour and shape features. Finally, customers express dissatisfaction regarding the misclassification of fruits.Therefore, the proposed methodology effectively addresses these limitations.

3. System Model

The CAD system under consideration encompasses several distinct stages: (a) preprocessing, aimed at improving the quality of the images; (b) segmentation, feature extraction, and optimisation, with the objective of selecting the most relevant data samples; and (c) classification of mango images into two categories: good or defective. The proposed methods are outlined in Figure 1.

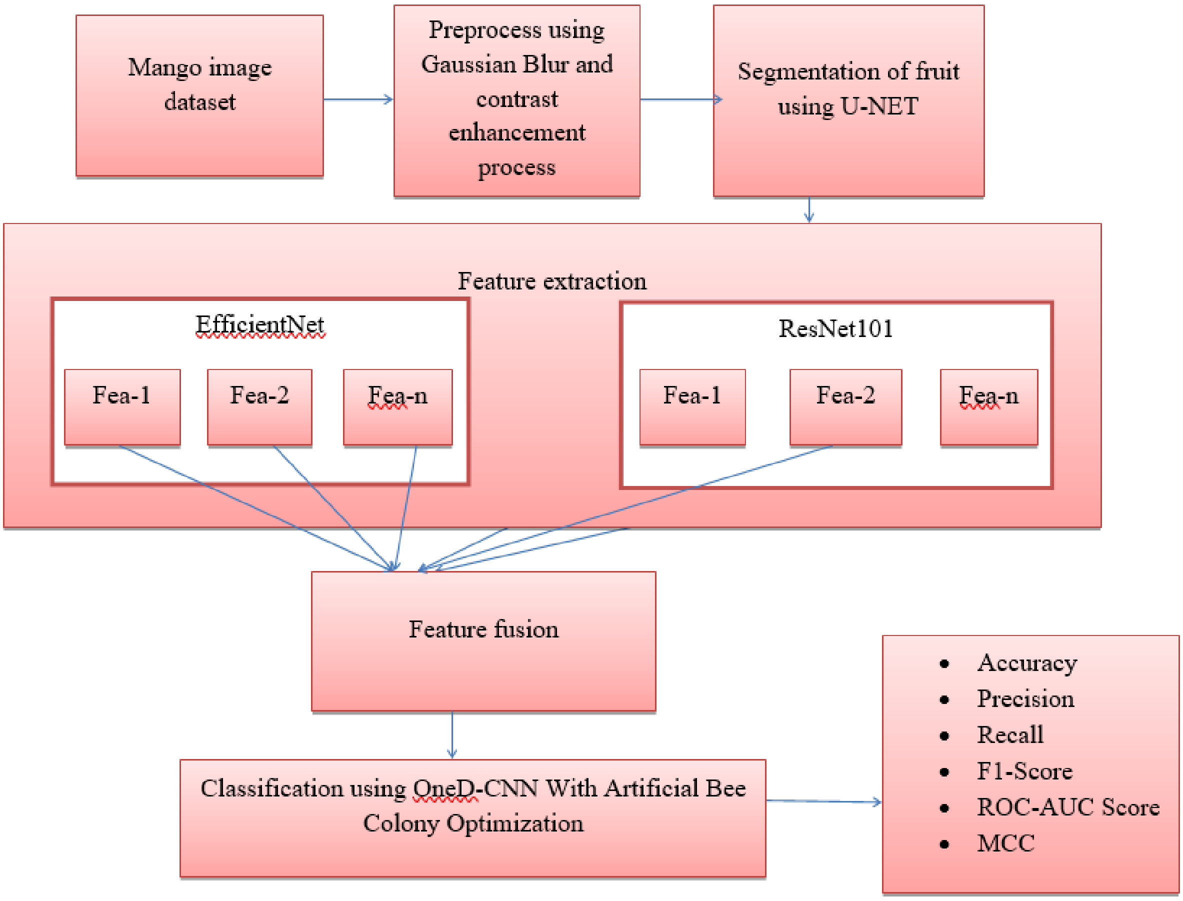

Mango image dataset
Preprocess using Gaussian Blur and contrast enhancement process
Segmentation of fruit using U-NET
Feature extraction
EfficientNet
Fea-1
Fea-2
Fea-n
ResNet101
Fea-1
Fea-2
Fea-n
Feature fusion
Classification using OneD-CNN With Artificial Bee Colony Optimization
Accuracy
Precision
Recall
F1-Score
ROC-AUC Score
MCC

Fig-1 system architecture for mango fruit classification

a. Preprocessing of images

Preprocessing constitutes a crucial stage within computer vision systems, wherein image quality is improved through various techniques and methods. The present research considers preprocessing techniques such as noise removal and image enhancement to effectively display areas of defects on the surface of fruits. The Gaussian Blur technique is employed in this work for the purpose of efficient noise removal. By applying image blurring techniques, the preservation of the low spatial frequency components of the image is achieved, while simultaneously reducing the presence of noise by eliminating extraneous details within the image.The Gaussian blur formula is represented by Equation (1), where *X* denotes the direction of the run being performed at a given time [24], and σ represents the standard deviation of the Gaussian distribution. The utilisation of the two-pass method for Gaussian filtering necessitates a reduced number of calculations, making it a more desirable choice for implementation.

$$g(X) = \frac{e^{-\frac{x^2}{2\sigma^2}}}{\sqrt{2\pi\sigma^2}} \quad (1)$$

The parameter σ governs the dispersion of data points around the mean value within the Gaussian distribution, thereby influencing the degree of blurring observed around a given pixel. We conducted experiments with a range of sigma values spanning from 0.1 to 16. It was observed that as the sigma value increased, there was a corresponding decrease in the high-frequency information content surrounding the pixel. Following the removal of noise, it is necessary to enhance the contrast, a task accomplished through the utilisation of an adaptive contrast enhancement method. Contrast refers to the disparity observed in the levels of pixel intensity, specifically between the highest and lowest values. The formula utilised for the purpose of stretching the histogram of an image in order to enhance the contrast is

$$g(x,y) = \frac{f(x,y)-fmin}{fmax-fmin} * 2^{bpp} \quad (2)$$

The formula necessitates the determination of the minimum and maximum pixel intensity, which is then multiplied by the levels of grey. In the present scenario, the image is encoded with 8bpp, resulting in a total of 256 possible levels of grey. The minimum value is zero, while the maximum value is two hundred and twenty-five. The formula is

$$g(x,y) = \frac{f(x,y)-0}{255-0} * 255 \quad (3)$$

The function f(x,y) represents the numerical value of the intensity of each pixel. The formula will be computed for each f(x,y) in the image. The application of mathematical equations results in the enhancement and improvement of the image's quality.The primary aim of the contrast enhancement system encompasses two main objectives: the implementation of locally adaptive histogram equalisation and the mitigation of undesired elements, such as noise and blocking objects. In this case, the block-based processing technique incorporates local adaptivity. The reduction of blocking artefacts is achieved by overlapping adjacent blocks, while noise is suppressed through the use ofspatio-temporally adaptive filtering.

b. U-NET based segmentation

The segmentation procedure may now begin on the preprocessed picture. As seen in figure-2, the U-NET architecture is used for segmentation. This model is somewhat similar to the auto-encoder in that it largely consists of two primary routes. The left path, often called the contracting or compressive path, is built like a regular CNN and serves as the encoder. The second route in the network is the decoder or the expanding path (or up-sampling or synthesizing path, depending on your source of information). Layers that perform deconvolution and convolution make up this section of the network. The suggested method downsamples the input pictures on the contracting path and then uses optimized approaches such concatenating skip connections on the expanding path to restore the original resolution and spatial structure of the images. The network learns geographic categorization information as it grows by making more precise and dense predictions. To top it all off, this procedure improves the resolution of the output, which is then passed on to the final convolutional layer to produce a segmented picture with the same dimensions as the input image.In this case, the network is tasked with taking an input picture of dimensions (h,w,n) and generating an output image of the same dimensions (h,w,n) in which the segmented region is highlighted. This requires doing things like without distorting the supplied image too much.

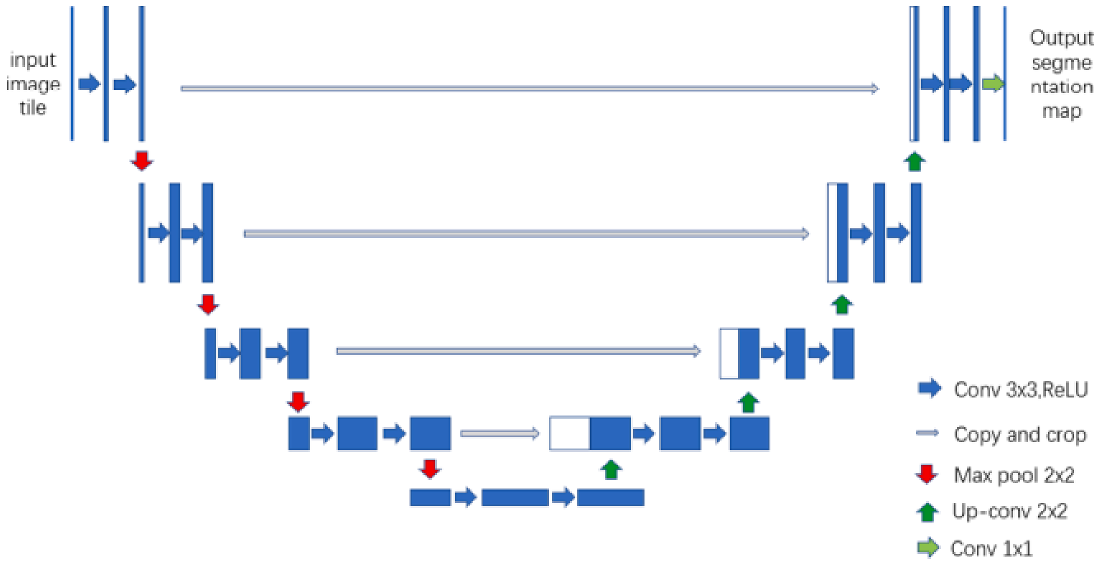


Fig-2 architecture of U-NET

The energy function is formulated by applying a pixel-wise soft-max operation to the final feature map, and then combining it with a cross entropy loss function. The softmax function is mathematically defined as follows:

$$p^k = exp(a_k(x)/(\sum_{k'=1}^{K} exp(a_k(x))$$
(4)

The activation in feature channel k' at each pixel position is denoted as $a_k(x)$, where K represents the number of classes and $pk(x)$ represents the approximated maximum-function. The formula for the weighted penalty is:

$$e = \sum \omega(x) log(p_{l(x)}(x)) \qquad (5)$$

Where l is the true label of each pixel and ω is a weight map. The weight map is then computed

Where$l$ represents the true label assigned to each pixel, while ω denotes the weight map. The computation of the weight map is

$$\omega(x) = \omega_c(x) + \omega_0. exp(-\frac{(d_1(x)+d_2(x))^\wedge 2}{2\sigma^2}$$
(6)

where weight map ω_c is employed to achieve class frequency balance. The variable d1 represents the distance to the nearest cell border, while d2 represents the distance to the second nearest cell border. The dice loss can be computed as

$$1 - 2(\sum_k x_k t_k)/(\sum_k x_k^2 + \sum_k t_k^2)$$
(7)

The dice loss variation where k iterates over all 512 × 512 pixels. Additionally, $\{x_k\}_k$ are FRUIT outputs (gray-scale images), taking values between 0 and 1 after going through the sigmoid function: $(1/(1 + e^{-x})$, while $\{t_k\}_k$are ground truth (masks), taking values of only 0 or 1, where 0 and 1 correspond to the background and the central roundish object, respectively; we swapped the roles and took the average.

In addition, the outputs of FRUIT, denoted as $\{x_k\}_k$, represent gray-scale images and are constrained to values ranging from 0 to 1. These outputs are obtained by passing the input through the sigmoid function: $(1/(1 + e^{-x})$ and the ground truth values, denoted as $\{t_k\}_k$, represent masks and are limited to binary values of 0 or 1. In this context, 0 and 1 correspond to the background and the central roundish object, respectively. In order to conduct data analysis, we performed a role exchange between the outputs and the ground truth values, followed by the calculation of their mean.Finally, the Intersection over Union (IoU), also known as the Jaccard index, can be expressed as:

$$IOU(A, B) = \frac{|A \cap B|}{|A \cup B|}$$
(8)

proximity quantifies the degree of similarity between two sets. The variable takes on values within the range of 0 to 1. When $A \cap B = \emptyset$, the variable is assigned a value of 0. Conversely, when $A = B$, the variable is assigned a value of 1. The function in question does not qualify as a loss function due to its lack of differentiability. However, it was employed in this paper to facilitate a comparison between masks generated by the FRUIT algorithm and the ground truth masks, thereby evaluating the performance of FRUIT.

c. Feature extraction of segmented fruit images

The process of feature texture extraction involves utilising image processing techniques to obtain the necessary features required for the subsequent classification phase. In this research, the grading of mango fruit based on its quality is conducted using a combination of colour, shape, and size features. To achieve this, an ensemble feature extractor comprising of EfficientNet and ResNet101 models is employed, along with a feature fusion process. EfficientNet, as a fundamental architecture, consists of a feature extraction model that incorporates a convolution layer followed by 16 MobileNet inverted bottleneck MBConv layers. These layers have dimensions of either 3x3 or 5x5. Subsequently, convolution, pooling, and fully connected layers are applied to the extracted features.EfficientNet is characterised by a relatively low number of trainable parameters, specifically 5.3 million, thereby facilitating more streamlined and efficient training processes.The MBconv block is a type of inverted residual block that incorporates layers for widening and expanding channels. In this design, direct connections are established between bottlenecks, which connect a smaller number of channels compared to the expansion layers. Additionally, this architectural design incorporates the utilisation of in-depth separable convolutions, which effectively reduce computational complexity by a factor of approximately k^2 when compared to conventional layers. Here, k denotes the kernel size.

To enhance performance and facilitate the utilisation of inverted residual connections, the Swish activation function was employed as a substitute for the ReLU activation function. The baseline network underwent a compound scaling factor to uniformly adjust the dimensions of depth, width, and resolution.In order to construct the variations of the baseline model, alternative values were employed to replace the compound coefficientpsi. The equation representing the relationship between the scaling of depth, width, and resolution is

$$depth: d = \propto \varphi \qquad (9)$$

$$width: w = \beta\varphi \qquad (10)$$

$$resolution: r = \gamma\varphi \quad (11)$$

$$\propto \geq 1, \beta \geq 1, \gamma \geq 1 \quad (12)$$

Where constants α, β, γ are determined through the process of grid search. The coefficient φ is a parameter specified by the user, which governs the quantity of resources available for scaling the model. Simply said, α, β and γ determine the network's breadth, depth, and resolution. ResNet101 is a popular computer vision convolutional neural network design. It addresses deep neural network vanishing gradients. The architecture consists of a total of 104 convolutional layers, organised into 3x3 blocks. Out of these blocks, 29 are directly connected to preceding blocks. Consider $D_s = \{(\gamma_1^s, \rho_1^s), \ldots. (\gamma_i^s, \rho_i^s), \ldots.. (\gamma_n^s, \rho_n^s)$with a learning task $L_D, L_S, (\gamma_m^s, \rho_m^s) \in R$; target domain $D_T = \{(\gamma_1^T, \rho_1^T), \ldots. (\gamma_i^T, \rho_i^T), \ldots.. (\gamma_m^T, \rho_m^T)$with a learning task $L_T,, (\gamma_n^s, \rho_n^s) \in R(m, n)$, is the training data sizes where $n \ll m$ and ρ_1^D and ρ_1^Tbe the labels of training data. Transfer learning is defined as:

$$D_s \neq D_T \quad (13)$$

The weights of the original models are transferred during the training of the new modified model. In the redesigned model, features are taken out of the feature layers of size N X 2048.

d. Fusion process

We next apply a Fuzzy Logic-based Low-Frequency Domain Fusion Rule to the retrieved features. In this research, greyscale picture data was used to fuzzy up the pixel points and classify them into five distinct categories. Then, the degree of membership of each fuzzy subset in the area of interest was determined using a triangle membership function. With this information in hand, fusion rules might be created, resulting in the aforementioned findings [25]. Cai and Wei started by breaking down the original image into its component frequencies. Following this, they developed fuzzy logic criteria for optimizing the data contained inside the fusion sub-band pictures and implemented the rules for low-frequency fusion [26]. This research uses the following definition of the Gaussian membership function to establish the weight coefficient of image fusion:

$$\mu_1(i, j) = exp\left[\frac{(f_1(i, j) - \pi)^2}{2(k\sigma)^2}\right] \quad (14)$$

where σ represents the standard deviation of both feature extractors. $f_1(i, j)$ denotes the low-frequency decomposition coefficient of a specific point,i, j, μ represents the average value of the decomposition coefficient. Additionally, k is a constant in the equation.

I. Classification using 1D-CNN with ACO

The ACO algorithm will be used to choose features when the fusion process is complete. Graph representation is a necessary step in the problem-representation process. Features are represented as nodes, and the edges between them show how one trait led to the selection of another. To determine the optimal collection of characteristics, an ant algorithm is used to search across a network, with the goal of making as few stops as possible at nodes that satisfy the stopping requirement.Let's say an ant is now stationed at node a and must decide which of the dotted-lined features it will use to continue on its journey.The rule for making these sorts of choices then moves on to selecting feature B, then c, and lastly d.When we get to node d, we know that the current subset, consisting of nodes a, b, c, and d, meets the traversal stopping condition (i.e., we've achieved a high enough level of classification accuracy). When the ant's search is complete, it will report this subset of features as a possible candidate for data compression. The likelihood that an ant at feature i will choose to migrate to feature j at time t is expressed by the so-called probabilistic transition rule, which is generated by combining the heuristic attractiveness of traversal with edge pheromone levels.

$$P_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)^{\alpha}.[\mu_{ij}]^{\beta}}{\sum_{I\epsilon j_i^k}[\tau_{il}(t)^{\alpha}.[\mu_{ij}]^{\beta}} & if\ j\epsilon j_i^k \\ 0 & otherwise \end{cases} \quad (15)$$

Where k is the total number of ants, i is a node in the graph, j is the set of nodes neighboring i that ant k has not yet visited, and j_i^k indicates the heuristic desirability of picking feature j when feature i is optional yet typically essential for getting a high algorithm performance.

The technique begins with the generation of k ants, which are then dispersed arbitrarily around the graph (with each ant initially equipped with a single random trait). You may also have as many ants as there are features in your data, with each ant starting the process of route building at a different feature.These nodes then begin their probabilistic edge exploration from these seed nodes until a stop condition is met. The resultant subsets are compiled and evaluated by us. If an optimal subset has been identified or the algorithm has been run a maximum number of times, the operation terminates and reports the best feature subset discovered.A new generation of ants is spawned, the pheromone is updated, and the process begins again if none of the requirements hold. Each edge's pheromone is kept up-to-date using the following formula:

$$\tau_{ij}(t+1) = (1-\rho).\tau_{ij}(t) + \rho.\Delta\tau_{ij}(t) \quad (16)$$

Where,

$$\Delta\tau_{ij}(t) = \sum_{k=1}^{n}(\gamma'(S^k)/|S^k| \quad (17)$$

If the edge $(i;j)$ has been traversed, then this is the case; otherwise, $\Delta\tau_{ij}(t)$ is 0. The feature subset discovered by ant k is S^k, and the number $0 \leq \rho \leq 1$ is the decay constant used to mimic the pheromone's evaporation.The size of the feature subset γ' itself and the "goodness" of the subset as a whole areconsidered while updating the pheromone.

This means that every ant updates the pheromone. The features are sent to 1D-CNN after being chosen. The nonlinear operation ReLU functions as an activation mechanism. The pool size and stride are both set to two for each max-pooling layer. The dropout layer is created using the parameter 0.4. The layers are made up of many neurons. This contains connections between the neurons in each layer. The last fully connected layers, which also provide the result layer, stand in for the prediction. The goal of convolutional neural networks is to convert image input into an output variable. When a prediction problem including characteristics as an input arises, they are considered as the go-to solution. The classification of the CNN structure considered each neuron's input, which is related to the local receptive field of the preceding layer and has a propensity to recover the local feature. Convolutional layer is considered to be the most important part of a CNN. The kernel in the layers settings has the least responsive deflect but enlarges across the whole of the input material. The output becomes more effective we add more convolutional layers since each one decreases the amount of input characteristics to fully connected layers, as seen in figure 3.

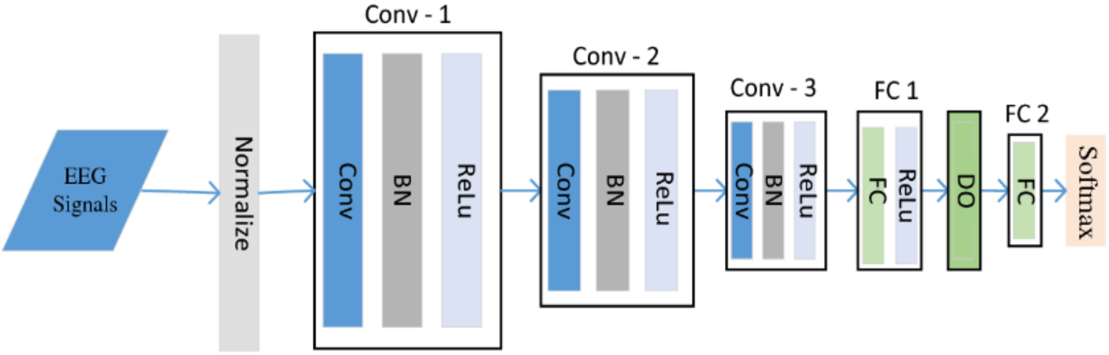


Fig-3. ID-CNN

Consider the input signal sequence as s'i and the filter as f_i where $i = 1,2,3, \dots . m$ and $i = 1,2,3, \dots . ni$. In this case, the length of the filter m must be longer than the length of the signal sequence ni.The filter is applied based on the input properties of the layer before it using a partial convolution operation. The 1d-CNN's convolved output, x_i, is

$$x_i = \sum_{g=1}^{m} f_g \times s_{i-g+1} \quad (18)$$

Here, the local window's $(d-1)^{th}$ layer of neurons and each neuron in the d^{th} layer are correlated to form the local connection network. The non-linear mapping is carried out via the activation function $af(s)$ in the convolution layer.

In the proposed 1D-CNN model, a modified linear unit incorporates an activation function to enhance the rate of convergence is

$$af(s') = \max(0, s') \quad (19)$$

Furthermore, the equation for the input of the q^{th} neurons in the d^{th} layer is obtained from equations 21 and 20.

$$c_q^d = af(\sum_{r=1}^{m} f_r^d \times c_{q-r+m}^{d-1} + h_d) \quad (20)$$

$$= af(f^d \times c_{(q-m+1):q}^{d-1} + h_d) \quad (21)$$

The offset parameters are $= 1,2,3, \dots . ni$, and the m^{th} dimension filter is $f^d €R^m$, which is the same for all convolution layer neurons. 1D-cnns perform well in fruit classification because of their streamlined array operations, simple training, and implementation with few hidden layers. The outputs from the 1D Convolution layer were utilised by the second CNN layer or block. The output of the second CNN layer has been subjected to the earlier Conv1D configuration and BM. After the second CNN layer, MaxPooling1D was launched with a sliding window that was three heights tall. Conv1D was altered as a result of the third CNN layer. Next, a 0.5-rate dropout layer was investigated to avoid overfitting. In order to construct a probability distribution over the two classes and categorise the data, a fully connected dense layer with softmax activation was used.

A. Convolutional layer (Maxpool-1D):

In neural networks, the 1D max pooling layer is a method for reducing the dimensionality of a feature map. This is accomplished by employing a predefined pool size and only maintaining the maximum value within a certain window patch. It receives the convolution outputs (c) from the conv1D layers. The convolution technique shifts/strides the window across the feature map. Max pooling works as

$$c_h^l = max c_p^{l-1}.r_h \qquad (22)$$

$r_h$is the pooling area with index h. The max pooling process is shown with these parameters: $c_{m1} = \max(c_1, c_2)$; $c_{m2} = \max(c_3, c_4)$; $c_{m3} = \max(c_5, c_6)$.

B. Flatten layer and dropout

The input data must be transformed into a one-dimensional vector by the flatten layer in order to be fed to the fully connected/dense layer. A dropout parameter is incorporated subsequent to the flatten layer in order to enhance the generalization capability of the architecture by mitigating the issue of overfitting during the training phase. Stochastically giving a value of zero to certain nodes' activations, as defined by a specified dropout rate, achieves the above mentioned goal. A 0.25 dropout rate has been used in this investigation.

C. Dense layer

The next layer, which is the dense or completely linked layer, receives the output that has been flattened as an input. This layer generates the classification output with a dimension of $M \times$, where M represents the total number of classes. In a general context, the layer operation can be expressed as

$$output = \sigma(< input, w_d > +b_d) \qquad (23)$$

where the term ($< input, w_d >$ represents the dot product between the weight vector w_d utilised in this layer and the input. Additionally, $b_d$is the bias vector, and σ isthe activation function. We employ sigmoid and softmax activation functions for the purpose of binary and multi-class classification, correspondingly. The sigmoid activation function is:

$$\sigma(z) = \frac{1}{1+e^{-z}} \qquad (24)$$

The function gives a probability value for binary categorization. This probability value determines a class label, either '0' or '1'. The softmax activation function is also termed

$$Softmax(z)_i = p_i = \frac{\exp(z)_i}{\sum_{j=1}^{m} \exp(z)_j} \qquad (25)$$

z layers i-th element is zi. The numerator is normalised by dividing it by the total of all exponential terms from 1 to M to keep pi between 0 and 1. This layer generates multi-class categorical class labels.

4. Result and Performance Analysis

The efficacy of the proposed strategy for classifying mangoes was evaluated with a classification test and k-fold cross-validation. Using k-fold cross-validation, all images were used for model training and assessment. This was accomplished by randomly allocating data samples into k separate groups. k minus one folds were used for training, whereas fold 1 was utilized for testing. The process is repeated for the remaining k1 folds. A 10-fold cross-validation method was utilized in this investigation. A defective mango will receive a positive result in the categorization evaluation, whereas a mango with no defects would receive a negative result. AUC, accuracy, precision, recall, f1-score, and Matthews Correlation Coefficient (MCC) were used to evaluate the classification performance.

Dataset description- [26]The dataset known as Fruit-360 consists of a total of 81 unique classes, where each class corresponds to a specific kind of fruit. The dataset has been partitioned into three distinct subgroups: the training set, which consists of 41,322 images; the validation set, which includes 9,744 images; and the testing set, which contains 4,133 images. In total, the dataset encompasses 55,244 images. The fruits were subjected to rotational motion using a low-speed motor operating at 3 rpm, and a 20-second video was captured to recorded.The background of all images was altered to a uniform white colour due to the presence of varying lightning conditions, which resulted in a lack of uniformity. Out of a total of 81 fruit categories, only a single image was obtained for each category. The dimensions of all images were standardised to 100 × 100 pixels, and a white background was applied. The use of high-resolution images is crucial for accurate image classification, as various fruits may possess similar colour and shape characteristics, while differing in size. Additionally, it is worth noting that certain fruits may exhibit similarities in terms of size, colour, and shape, despite belonging to distinct categories. For instance, apples, mangoes, and tomatoes possess numerous variants within their respective types. In order tocategorise fruits of this particular type, various variations of the same fruit were organised into distinct classes. In contrast to other general classification datasets, the Fruit-360 dataset exclusively comprises images of fruits. This dataset is characterized by significant intra-class similarities, as the fruits share common features.

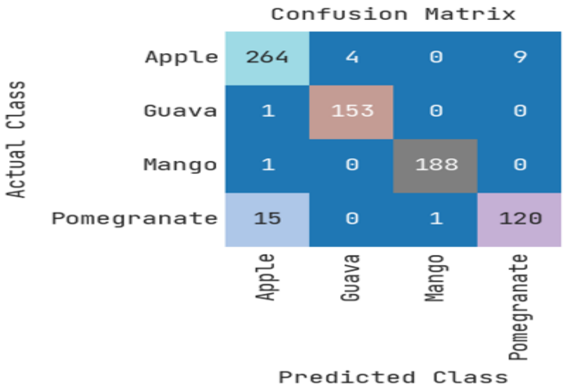


Fig-4 confusion matrix of ACO_1DCNN for testing

The confusion matrix for features using testing for ACO_1DCNN is shown in Figure 4 above, where the rows and the columns showsthe predicted and actual class of data relevant to the prediction of mango fruit. The tested networks that are correctly and wronglycategorised are shown by the crosswisecolours. The row below shows the execution of each real class, whereas the column on the right shows each anticipated class.

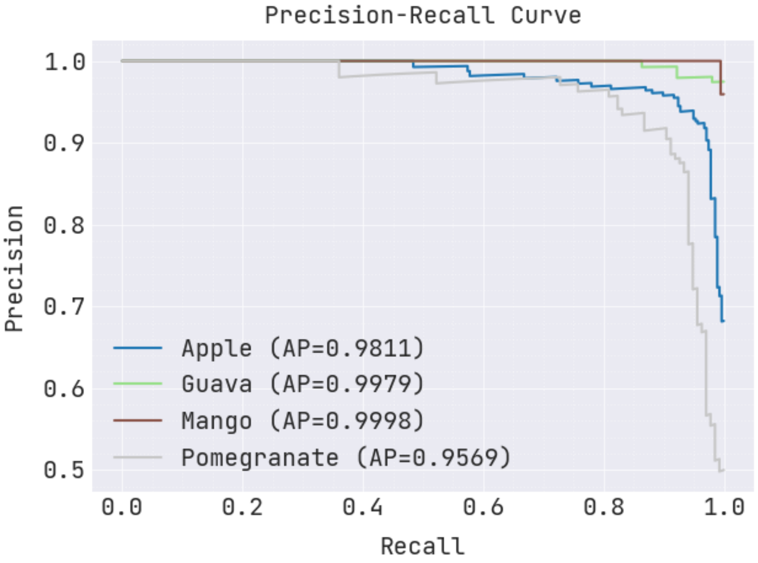


Fig-5 precision-recall curve of ACO_1DCNN for testing

The above figure 5 shows the precision-recall curve for testing ACO_1DCNNwhere the recall value is shown on the x-axis and the accuracy value is shown on the y-axis. During the process, the AP achieves maximum AP as, 0.9811 for apple, 0.9979 for guava, 0.9998 for mango and 0.9569 for pomegranate.

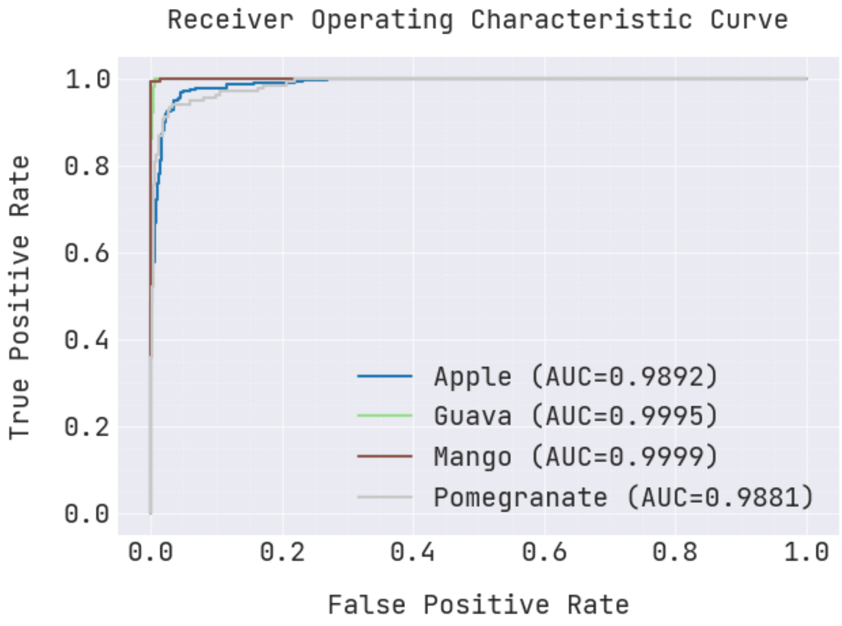
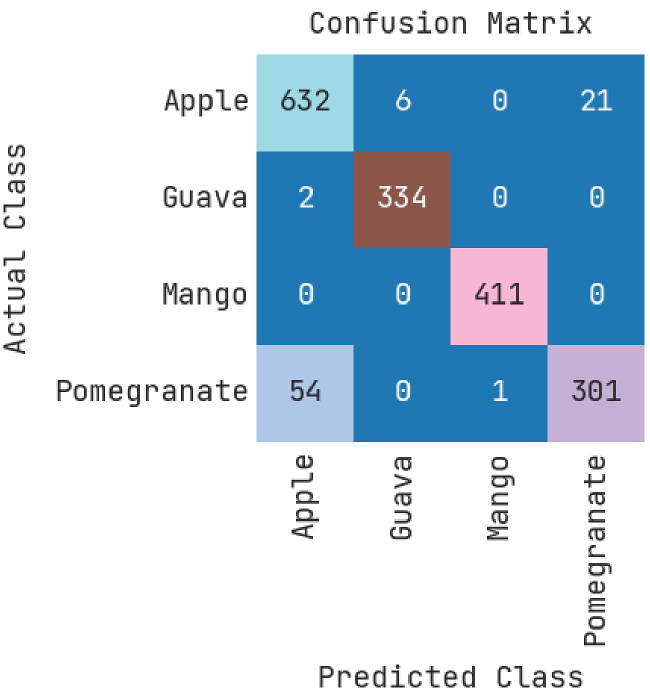


Fig-6 ROC curve of ACO_1DCNN for testing

Figure 6 shows ROCcurve for testing ACO-1DCNNwhere the true positive rate and false positive rate are shown by the x- and y-axes, respectively. During the process the AUC achieves maximum range of 0.9892 for apple, 0.9995 for guava, 0.999 for mango and 0.9881 for pomegranate



	Apple	Guava	Mango	Pomegranate
Apple	632	6	0	21
Guava	2	334	0	0
Mango	0	0	411	0
Pomegranate	54	0	1	301

Fig-7 confusion matrix of ACO_1DCNN for training

The confusion matrix for features using training for ACO_1DCNN is shown in Figure 7 above, where the columns provide the actual class of data important to forecasting mango fruit and the rows display the anticipated class. The tested networks that are correctly and wronglycategorised are shown by the crosswisecolours. The row below shows the execution of each real class, whereas the column on the right shows each anticipated class.

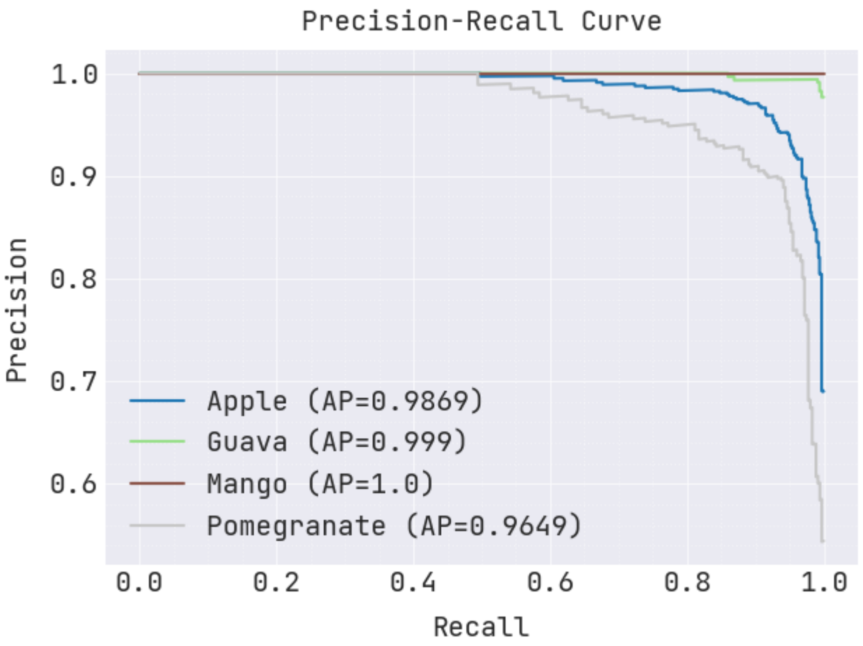


Fig-8 precision-recall curve of ACO_1DCNN for training

The above figure 8 shows the precision-recall curve for training ACO_1DCNNwhere the recall value is shown on the x-axis and the accuracy value is shown on the y-axis. During the process, the AP achieves maximum AP as, 0.9869 for apple, 0.999 for guava, 1.0 for mango and 0.9649for pomegranate.

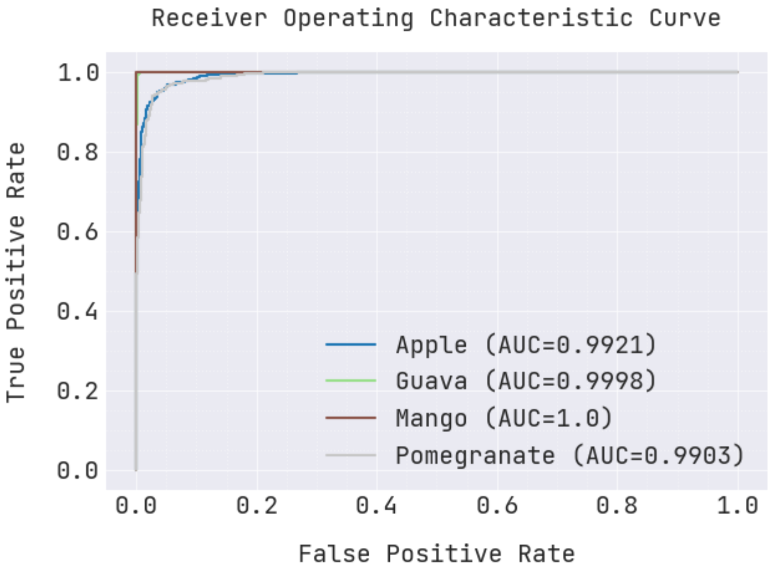


Fig-9 ROC curve of ACO_1DCNN for training

Figure 6 shows ROC curve for training ACO-1DCNNwhere the true positive rate is shown on the y-axis and the false positive rate is shown on the x-axis. During the process the AUC achieves maximum range of 0.9921 for apple, 0.9998 for guava, 1.0 for mango and 0.9881 for pomegranate

Table-1 analysis of various metrics on ACO-1DCNN for testing and training

Metrics	Testing	Training
Accuracy (%)	97.95	97.62
Precision (%)	95.97	95.83
Recall (%)	95.59	94.96

F-Score (%)	95.76	95.33
ROC AUC Score (%)	99.42	99.56
MCC (%)	94.34	93.69

The above Table -1 determines the various metrics of the proposed technique (ACO-1DCNN) and the Table-2 determines the Overall Comparative Analysis of the Existing technique such as the CNN,Resnet152,InceptionV3,VGG16 and the proposed technique ACO-1DCNN.

Table -2 Overall comparative Analysis of the Existing and Proposed Technique

	CNN	Resnet152	Inception V3	VGG16	ACO-1DCNN
Accuracy(%)	88.65	93.43	95.45	96.18	97.95
Precision(%)	85.65	87.15	90.17	93.44	95.97
Recall(%)	84.13	86.1	89.23	93.15	95.59
F-Score(%)	88.14	91.33	93.17	94.18	95,76
ROC AUC Score(%)	90.12	92.15	94.43	97.11	99.42

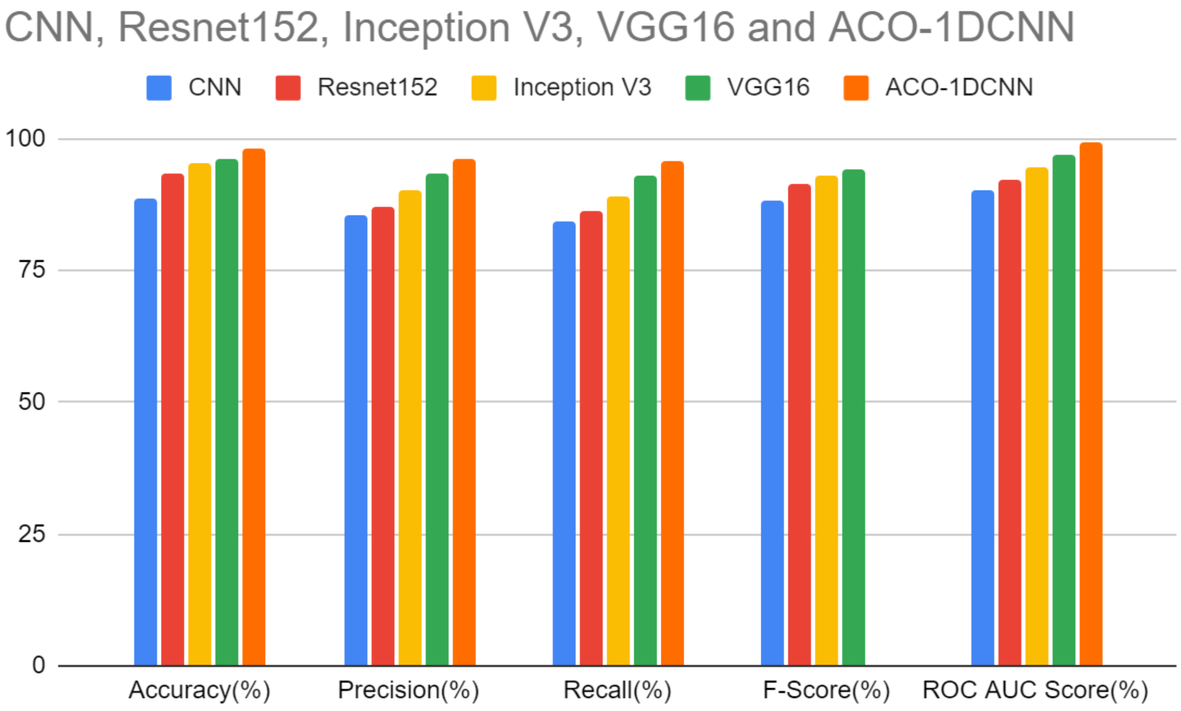
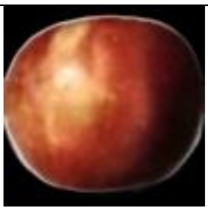
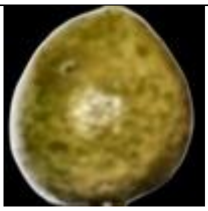
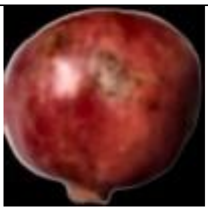


Fig.10. Overall performance analysis of the Existing Technique with the Proposed technique(ACO-1DCNN)

The above Fig 10 determines the Overall performace analysis of the Existing technique such as the CNN,Resnet152,Inception V3 and VGG 16 with the proposed technique ACO-1DCNN ,where the proposed technique has the high level of Accuracy of 97.95% ,Precision of 95.97%,Recall of 95.59%,F-Score 95.76%,ROC AUC 99.42%.

Table-2 Various Processed Image of Fruit by Proposed Technique

Input Image	Pre-processed	Segmented	Output Image	Un-healthy
				
				
				

The above Table-2 portrays the various processed Image of Fruit by Proposed Technique ACO-1DCNN and the table identifies the healthy and un-healthy fruit

5. Conclusion

This paper presents a novel approach utilising ACO-1DCNN for the classification of mango quality, specifically in the presence of interfering background elements. The model under consideration was trained and validated, achieving a training and validation accuracy of 97.62%. The model under consideration exhibited the most efficient training times. The findings of this research indicate that the proposed model exhibits significant promise in the realm of mango quality detection and classification. In the future, this technology will have the capability to identify additional quality attributes of mangoes, such as colour, size, types, ripeness, and physiological disorders. The proposed model has the potential to be expanded to encompass a greater number of mango quality categories and to classify other types of fruits. Furthermore, it is anticipated that the model's performance will be evaluated in the future through deployment on real-time online sorting equipment.

Reference

[1] D. Pem and R. Jeewon, "Fruit and vegetable intake: benefits and progress of nutrition education interventions- narrative review article," *Iranian Journal of Public Health*, vol. 44, no. 10, pp. 1309–1321, 2015.

[2] R. C. Fierascu, E. Sieniawska, A. Ortan, I. Fierascu, and J. Xiao, "Fruits by-products - a source of valuable active principles. a short review," *Frontiers in Bioengineering and Biotechnology*, vol. 8, pp. 319–328, 2020.

[3] S. R. Dubey and A. S. Jalal, "Species and variety detection of fruits and vegetables from images," *International Journal of Applied Pattern Recognition*, vol. 1, pp. 108–126, 2013.

[4] R. A. A. Al-Fallujah, “Color, shape, and texture based fruit recognition system,” *International Journal of Advanced Research in Computer Engineering & Technology*, vol. 5, pp. 2108–2112, 2016.

[5] S. Arivazhagan, S. Newlin, N. Selva, and G. Lakshmanan, “Fruit recognition using color and texture features,” *Journal of Emerging Trends in Computing and Information Sciences*, vol. 1, no. 2, pp. 90–94, 2010.

[6] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, et al., "Backpropagation applied to handwritten zip code recognition," Neural computation, vol. 1, pp. 541-551, 1989.

[7] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in Advances in neural information processing systems, 2012, pp. 1097-1105.

[8] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, 2014, pp. 1725-1732.

[9] Y. Sun, Y. Chen, X. Wang, and X. Tang, "Deep learning face representation by joint identification-verification," in Advances in neural information processing systems, 2014, pp. 1988-1996.

[10] F. Siddique, S. Sakib, and M. A. B. Siddique, "Handwritten Digit Recognition using Convolutional Neural Network in Python with Tensorflow and Observe the Variation of Accuracies for Various Hidden Layers," 2019.

[11] I. Sa, Z. Ge, F. Dayoub, B. Upcroft, T. Perez, and C. McCool, "Deepfruits: A fruit detection system using deep neural networks," Sensors, vol. 16, p. 1222, 2016.

[12] D. H. Hubel and T. N. Wiesel, "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex," The Journal of physiology, vol. 160, pp. 106-154, 1962.

[13] K. Fukushima and S. Miyake, "Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition," in Competition and cooperation in neural nets, ed: Springer, 1982, pp. 267-285.

[14] C. Hung, J. Nieto, Z. Taylor, J. Underwood, and S. Sukkarieh, "Orchard fruit segmentation using multi-spectral feature learning," in 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2013, pp. 5314-5320

[15] Sakib, S., Ashrafi, Z., & Siddique, M. A. B. (2019). Implementation of fruits recognition classifier using convolutional neural network algorithm for observation of accuracies for various hidden layers. *arXiv preprint arXiv:1904.00783*.

[16] Fu, L., Gao, F., Wu, J., Li, R., Karkee, M., & Zhang, Q. (2020). Application of consumer RGB-D cameras for fruit detection and localization in field: A critical review. *Computers and Electronics in Agriculture*, *177*, 105687.

[17] Yeh, J. F., Lin, K. M., Lin, C. Y., & Kang, J. C. (2023). Intelligent Mango Fruit Grade Classification Using AlexNet-SPP With Mask R-CNN-Based Segmentation Algorithm. *IEEE Transactions on AgriFood Electronics*.

[18] Kumari, N., Bhatt, A. K., & Dwivedi, R. K. (2023, April). Self-Adaptive Glow Warm Swarm Optimization Technique in Optimal Feature Selection in Grading of Fruit Mango. In *2023 International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES)* (pp. 424-430). IEEE.

[19] Rajalaxmi, R. R., Saradha, M., Fathima, S. K., Sathish Kumar, V. E., Sandeep Kumar, M., & Prabhu, J. (2023). An Improved MangoNet Architecture Using Harris Hawks Optimization for Fruit Classification with Uncertainty Estimation. *Journal of Uncertain Systems*, *16*(01), 2242006.

[20] Gururaj, N., Vinod, V., & Vijayakumar, K. (2022). Deep grading of mangoes using Convolutional Neural Network and Computer Vision. *Multimedia Tools and Applications*, 1-26.

[21] Rizwan Iqbal, H. M., & Hakim, A. (2022). Classification and grading of harvested mangoes using convolutional neural network. *International Journal of Fruit Science*, *22*(1), 95-109.

[22] Kumari, N., Kr. Bhatt, A., Kr. Dwivedi, R., &Belwal, R. (2021). Hybridized approach of image segmentation in classification of fruit mango using BPNN and discriminant analyzer. *Multimedia Tools and Applications*, *80*, 4943-4973.

[23] Shi, R., Li, T., & Yamaguchi, Y. (2020). An attribution-based pruning method for real-time mango detection with YOLO network. *Computers and electronics in agriculture*, *169*, 105214.

[24] Pulfer, E.M. *Different Approaches to Blurring Digital Images and Their Effect on Facial Detection*; University of Arkansas: Fayetteville, NC, USA, 2019.

[25] J. Teng, S. Wang, J. Zhang, and W. Xue, "Fusion algorithm of medical images based on fuzzy logic," in *2010 Seventh International Conference on Fuzzy Systems and Knowledge Discovery*, pp. 546–559, Yantai, China, 2010.

[26] W. Cai and Z. Wei, "Pii GAN: generative adversarial networks for pluralistic image inpainting," *IEEE Access*, vol. 8, pp. 48451–48463, 2019.

[27] Kausar, A., Sharif, M., Park, J., & Shin, D. R. (2018, December). Pure-cnn: A framework for fruit images classification. In *2018 International Conference on Computational Science and Computational Intelligence (CSCI)* (pp. 404-408). IEEE.

[28] Prof. Vaishali Sarangpure. (2018). Hybrid Hand-off Scheme for Performance Improvisation of Wireless Networks. International Journal of New Practices in Management and Engineering, 7(03), 08 - 14. https://doi.org/10.17762/ijnpme.v7i03.67

[29] Martínez, L., Milić, M., Popova, E., Smit, S., & Goldberg, R. Machine Learning Approaches for Human Activity Recognition. Kuwait Journal of Machine Learning, 1(4). Retrieved from http://kuwaitjournals.com/index.php/kjml/article/view/146

[30] Joshi, K., Kumar, V., Sundaresan, V., Ashish Kumar Karanam, S., Dhabliya, D., Daniel Shadrach, F., & Ramachandra, A. C. (2022). Intelligent fusion approach for MRI and CT imaging using CNN with wavelet transform approach. Paper presented at the IEEE International Conference on Knowledge Engineering and Communication Systems, ICKES 2022, doi:10.1109/ICKECS56523.2022.10060322 Retrieved from www.scopus.com