computers

MDPI

Article

Botanical Leaf Disease Detection and Classification Using Convolutional Neural Network: A Hybrid Metaheuristic Enabled Approach

Madhumini Mohapatra [1], Ami Kumar Parida [1], Pradeep Kumar Mallick [2], Mikhail Zymbler [3] and Sachin Kumar [3,*]

[1] School of Engineering and Technology (ECE), GIET University, Gunupur 765022, Odisha, India; m_mohapatra@gcekjr.ac.in (M.M.); ami85parida@gmail.com (A.K.P.)
[2] School of Computer Engineering, Kalinga Institute of Industrial Technology (KIIT), Deemed to be University, Bhuvaneswar 751024, Odisha, India; pradeepmallick84@gmail.com
[3] Department of Computer Science, South Ural State University, Chelyabinsk 454080, Russia; mzym@susu.ru
* Correspondence: sachinagnihotri16@gmail.com

Abstract: Botanical plants suffer from several types of diseases that must be identified early to improve the production of fruits and vegetables. Mango fruit is one of the most popular and desirable fruits worldwide due to its taste and richness in vitamins. However, plant diseases also affect these plants' production and quality. This study proposes a convolutional neural network (CNN)-based metaheuristic approach for disease diagnosis and detection. The proposed approach involves preprocessing, image segmentation, feature extraction, and disease classification. First, the image of mango leaves is enhanced using histogram equalization and contrast enhancement. Then, a geometric mean-based neutrosophic with a fuzzy c-means method is used for segmentation. Next, the essential features are retrieved from the segmented images, including the Upgraded Local Binary Pattern (ULBP), color, and pixel features. Finally, these features are given into the disease detection phase, which is modeled using a Convolutional Neural Network (CNN) (deep learning model). Furthermore, to enhance the classification accuracy of CNN, the weights are fine-tuned using a new hybrid optimization model referred to as Cat Swarm Updated Black Widow Model (CSUBW). The new hybrid optimization model is developed by hybridizing the standard Cat Swarm Optimization Algorithm (CSO) and Black Widow Optimization Algorithm (BWO). Finally, a performance evaluation is undergone to validate the efficiency of the projected model.

Keywords: botanical leaf disease; Black Widow Algorithm; CNN; Cat Swarm Optimization

check for updates

Citation: Mohapatra, M.; Parida A.K.; Mallick, P.K.; Zymbler, M.; Kumar, S. Botanical Leaf Disease Detection and Classification Using Convolutional Neural Network: A Hybrid Metaheuristic Enabled Approach. *Computers* **2022**, *11*, 82. https://doi.org/10.3390/computers11050082

Academic Editor: Paolo Bellavista

Received: 20 April 2022
Accepted: 18 May 2022
Published: 20 May 2022

1. Introduction

Mango fruit is one of the most popular fruits worldwide due to its rich vitamins and fantastic taste. India is one of the largest producers of mangoes, and it produces 40% of the world's mangoes [1]. Pests and diseases damage crop production, crushing 30% to 40% of the crop yield [2]. Unaided eye vision is used to identify the mango plant pathogens, which has less precision. Farmers are unaware of the various diseases that affect mango plants, resulting in lower mango fruit yield. The various diseases wreak havoc on the mango harvest, and uneven colored black spots appear due to the disease. These patches occur on the leaf surface or young fruits [3–6]. These patches start small but soon spread to the whole fruit or leaves, causing the fruit to rot. These illnesses must be diagnosed and monitored in a specific time frame while they are still in their early stages [7–9]. These diseases are caused by pathogens such as champignons, bacteria, and viruses, resulting in plant death. Identifying plant diseases is the process of agricultural experts inspecting each plant regularly. Farmers must actively track the plant body for this, which is a time-consuming process [10–12]. Early identification of plant disease needs the use of different techniques. Early recognition of the disease in the field is the initial step in managing the spread of mango diseases. Traditional disease detection strategies rely on the help of agricultural organizations, but these methods are limited due to a lack of logistical and human resources [13–15]. Mobile phones and UAVs use technologies to increase internet access and have new instruments for disease detection that rely on automatic image recognition to aid in large-scale detection. With the introduction of CV, ML, and AI technologies, advancement has been achieved in building automated models that enable the accurate and timely diagnosis of plant leaf disease [16,17]. With the advent of a variety of high-performance computer processors and devices in the previous decade, AI and machine learning technologies have attracted interest. DL has been well acknowledged as being primarily employed in agriculture [18]. This idea seems crucial for establishing, regulating, sustaining, and improving agricultural productivity. It is at the core of the intelligent farming technique, which is identified to incorporate new technology, algorithms, and gadgets into farming [18,19]. Deep learning using Neural Networks is a part of machine learning. The advancement of such computer technologies will assist farmers in monitoring and controlling plant diseases. Previous research has shown that image recognition can be used to recognize plant disease in maize, apples [17–19], and other stable and diseased plants [20]. The detection of mango leaf diseases using automatic image recognition and attribute extraction has shown positive results [2,3]. However, extraction characteristics are computationally intensive and require solid performance expertise. Therefore, the optimized deep learning models are suggested as a promising solution. The significant contributions of this research work follow:

- Introduces a new geometric mean-based neutrosophic with fuzzy c-mean for segmenting the diseased region from the standard leaf regions;
- Extracts Upgraded Local Binary Pattern (ULBP) to train the detection model precisely, which results in the enhancement of texture features;
- Introduces a new optimized CNN model to detect the presence/absence of leaf disease in mango trees;
- Introduces a new hybrid meta-heuristic optimization model called Cat Swarm Updated Black Widow Model (CSUBW) to optimize the CNN.

This paper is organized as follows. Section 2 discusses the literature review. Sections 3 and 4 provides the detailed description of the proposed methodology. Section 5 provides the description of the experiments and results. This paper is concluded in Section 6.

2. Literature Review

Plant disease detection is critical for implementing disease management strategies that increase crop quality and quantity. Plant disease automation is advantageous since it decreases the monitoring required on big farms. As the leaves are a plant's primary source of nutrition, it is critical to identify leaf diseases early and precisely. Farmers and plant pathologists are implicated in traditional disease management methods. In the fields, pesticide diagnosis and application are more common. This approach is time consuming and challenging, and it frequently leads to erroneous diagnoses and inappropriate pesticide application [16]. Different authors have suggested several deep learning architectures throughout this era of study. CNN is one of the most often used deep learning algorithms among them. The biological nerve and visual systems inspired CNN. It is a deep learning classification model that is supervised and has a good classification and recognition accuracy. This paradigm has a complicated structure due to the massive number of information processing layers. The multiplayer architecture distinguishes itself from traditional Artificial Neural Networks (ANNs) [2]. They can learn features from a training dataset. Compared to standard ANN models, CNN models require many neurons but a massive amount of data to train [5,9]. Table 1 summarizes the characteristics and difficulties of the existing literature-based studies. In 2020, Chouhan et al. [1] suggested an automated web-based leaf disease segmentation approach based on an NN. The intended system was divided into four radial basis functions: First, using a web-enabled digital camera, images of mango leaves are taken in real time. Second, using a “scale-invariant feature transform technique”, the images were preprocessed, and features were extracted. Third, the bacterial foraging optimization technique has been used to optimize the NN’s training by utilizing the most distinct characteristics. Finally, the damaged region of the mango leaf pictures was extracted using the radial basis function NN. The testing findings confirmed the suggested system’s high accuracy in segmenting anthracnose (fungal) disease. In 2020, Mia et al. [2] proposed an NNE for MLDR to identify diseases quickly and correctly. Training data were created using a classification model that collects pictures of leaves that different diseases have damaged. An ML system was designed to autonomously upload and correlate fresh pictures of damaged leaves with training data to determine mango leaf diseases. With an average accuracy of 80%, the suggested system could correctly identify and categorize the investigated illness. This recommended remedy suffocated the mango plants. The technology had assisted in disease detection without any involvement of an agriculturist, thereby improving the efficiency by identifying disease areas with a machine learning model rather than a mechanical approach. It would also cure the afflicted mango leaf disease appropriately, boost mango output, and fulfill worldwide market demand. Venkatesh et al. [3] have proposed a “V2IncepNet” that integrates the best feature of the Inception module. The VGGNet module extracts basic features, whereas the Inception module does high-dimensional feature extraction and image categorization. The VGGNet module extracts essential functionality, whereas the Inception module does high-dimensional feature extraction and picture categorization. Certain color features are all used in this work. According to the results, the suggested model can classify the amount of Anthracnose disease infection on mango leaves with at least 92% accuracy. The suggested model was straightforward but effective. In 2020, Pham et al. [16] developed an ANN technique to identify early leaf tissue illnesses with tiny spots that can only be seen with higher resolution images. All infected blobs were segmented for the whole dataset after the preprocessing using a contrast enhancement technique. A list of various measurement-based features representing the blobs was picked using a wrapper-based feature selection technique based on a hybrid metaheuristic and then selected depending on their impact on the model’s performance. The chosen features were then transferred as inputs to the ANN. They evaluate the numbers achieved using their approaches to those acquired using a different strategy incorporating transfer learning and prominent CNN models (AlexNet, VGG16, ResNet-50). For the categorization of mango leaves affected by the fungal disease Anthracnose, researchers have suggested MCNN. This research was based on a real-time dataset of 1070 mango tree leaves recorded at “Shri Mata Vaishno Devi University in Katra, J&K, India”. Both the healthy and diseased leaf images were included in the dataset. Compared to other state-of-the-art methods, the suggested MCNN model appears to have a better classification accuracy. In 2017, Ullagaddi and Raju [19] proposed a directional feature extraction technique based on the MRKT to address difficulties with plant disease detection caused by shape, color, or other misleading features. The directional characteristics and histograms for plant components such as the leaf and fruit of the mango crop were calculated using an MRKT-based approach. These histograms and the directional features are set together with an Artificial Neural Network, resulting in a more accurate diagnosis of Anthracnose disease, which appears as black spots on mango fruits and leaves. The findings obtained by the suggested MRKT directional feature set have demonstrated that the proposed concept has produced better results with an accuracy of up to 98%. In 2021, Kumar et al. [20] developed a new CNN architecture to detect mango Anthracnose disease. Testing was performed using a “real-time dataset” collected from “Karnataka, Maharashtra, and New Delhi farms”. It includes both healthy and unhealthy photographs of mango tree leaves. In 2021, Sujatha et al. [4] used ANN to detect mango leaf diseases early. The digital camera was used to capture the image of the damaged leaf; the image was taken from a consistent distance with adequate illumination. The noise reduction using an averaging filter, color transformation, and histogram equalization has been used to preprocess the image captured with the digital camera. Compared to other image segmentation methods, histogram-based approaches were highly efficient, since they generally require one pass across the pixels. The k-means method was used to segment the images in this research. Many feature extraction approaches have been applied, including texture characteristics based on GLCM and SGDM. Texture features include contrast, energy, local homogeneity, cluster shadow, and prominence.

Table 1. Features and Challenges of Existing Works Focused on Mango Leaf Disease Detection.

Author	Adopted Methodology	Advantages	Drawbacks
Chouhan et al. [1]	Radial Basis Function (RBF) Neural Network	higher specificity and sensitivity, intuitive, user-friendly	need to overcome over-segmentation problem; highly prone to noise; not applicable in industrial applications
Mia et al. [2]	neural network	consumes less time	accuracy of classification is lower; risk of over-fitting; not used in industrial applications
Venkatesh et al. [3]	VGGNet model	simple and cost-effective; used in industrial applications	lower detection accuracy
Pham et al. [16]	Feed-Forward Neural Network and Hybrid Metaheuristic Feature Selection	higher testing accuracy, recall, precision and F1-score	higher computational complexity in terms of time and cost
Singh et al. [13]	Multilayer Convolution Neural Network	higher classification accuracy (97.13%) computationally efficient; used in different industrial conditions	highly prone to noise
Ullagaddi and Raju [19]	Modified Rotational Kernel Transform Features	higher reorganising accuracy and segmentation accuracy	lower miss rate, specificity and sensitivity; not applicable in industrial conditions
Kumar et al. [20]	CNN	higher classification accuracy; used in different industrial constraints	lower sensitivity and specificity higher misclassification
Sujatha et al. [4]	ANN	less prone to noise, efficient extraction metod92.31	higher computational complexity; not applicable in industrial conditions

3. Proposed Methodology

This study proposed a mango leaf detection approach that consists of the following stages: (a) preprocessing, (b) image segmentation, (c) feature extraction, and (d) disease prediction. The architecture of the proposed work is manifested in Figure 1. Initially, the collected raw image is preprocessed via "contrast enhancement and histogram equalization" to remove the noise and other unwanted artifacts to enhance the quality of the image. Then, preprocessed images are segmented via the proposed geometric mean-based neutrosophic with fuzzy C-mean. Subsequently, the most relevant features such as ULBP (texture feature), color feature, and pixel features are extracted from the segmented images. Finally, these features are fed as the input to the detection phase, which is modeled using a CNN (deep learning model) for disease identification. Furthermore, to enhance CNN's classification accuracy, its weights are fine-tuned by a new hybrid optimization model. The new hybrid optimization model hybridizes the standard CSO and BWO.

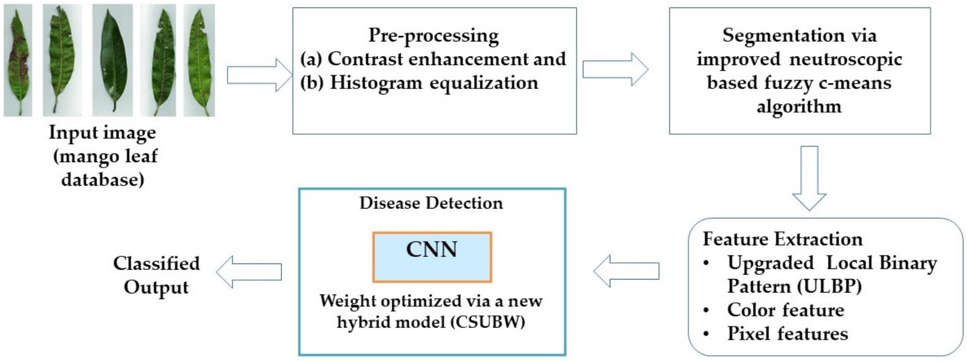

Figure 1. Architecture of the proposed work.

3.1. Preprocessing: Contrast Enhancement and Histogram Equialization

The phrase "image preprocessing" refers to actions on images at the most fundamental level. If entropy is an information metric, these techniques do not enhance picture information content but rather reduce it. Instead, preprocessing aims to enhance the picture data by suppressing unwanted distortions or enhancing important visual characteristics for subsequent analysis and processing. In this research work, the collected input image I^{in} is preprocessed via contrast enhancement and a histogram equalization model. The preprocessing phases are diagrammatically shown in Figure 2.

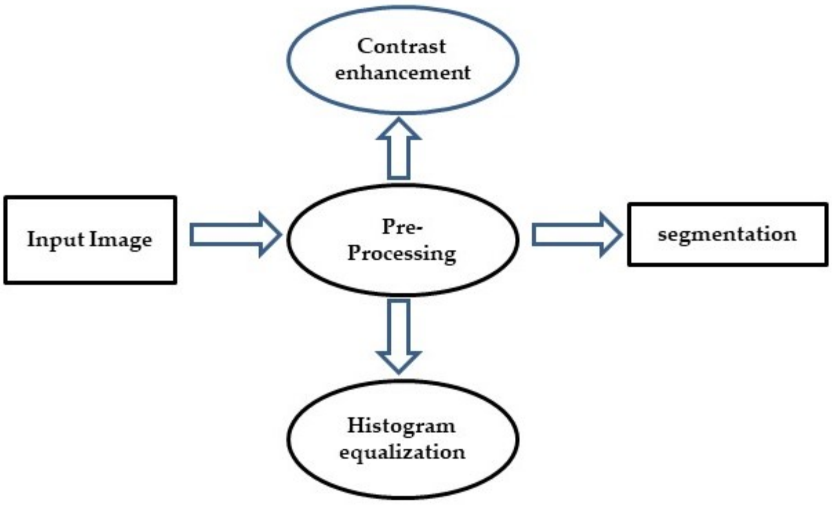

Figure 2. Preprocessing phase.

3.2. Contrast Enhancement

The quality of the input image I^{in} is enhanced by applying the contrast enhancement technique [21]. Initially, the RGB color channel is converted into HSI. This approach is centered around the intensity parameters and preserves the other hue and saturation values. Then, the intensity is separated into two-step parameter groups: low δ_{low} and high δ_{high} . These two groups can be expressed as per Equations (1) and (2), respectively.

$$\delta_{low} = \{\delta_j || j \leq \delta_m\} \tag{1}$$

$$\delta_{high} = \{\delta_i || i \ > \delta_m\} \tag{2}$$

Here, δ_m is the trivial threshold intensity value. Furthermore, the enhanced intensity is computed by using Equation (3).

$$\delta_{enhance}(i) = \delta_{low} + (\delta_{high} - \delta_{low}) \times \beta_i = \{\delta_i \,\| i > \delta_m\} \tag{3}$$

Here, β_i is the cumulative density computed from the histogram. Next, the mean brightness and the input brightness are calculated to reduce the inaccuracy. The repetition of this procedure continues until the appropriate increased intensity values are found. Finally, the result is created by combining the increased intensity with the other initial hue and saturation values and converting them to an RGB color channel.

3.3. Histogram Equalization

The intensity distribution of an image is graphically represented by a histogram [22]. It essentially indicates the number of pixels for every pixel intensity considered. Histogram equalization is a contrast-enhancing computer image processing method. It effectively spreads out some of the most common intensity values, i.e., widening the image's intensity range. When near contrast values represent the useable data, this approach generally boosts the global contrast of images and enables locations with poor local contrast to get a boost in contrasts. A color histogram indicates the pixel values inside each color image. Histogram equalization cannot be applied independently to the image's red, green, and blue components, since it causes drastic color shifts. However, suppose the picture is transformed to another color space, such as HSL/HSV. The method can be applied to the luminance or value channel without changing the image's hue or saturation. The preprocessed image acquired at the end of histogram equalization is denoted as I^{pre}, which is subjected to the proposed geometric mean with a modified fuzzy C-means-based neutrosophic segmentation phase.

3.4. Proposed Image Segmentation Phase

Geometric Mean with Modified Fuzzy C-Means Based Neutrosophic Segmentation Phase

The preprocessed image I^{pre} is segmented via geometric mean with fuzzy C-means-based neutrosophic segmentation. The universe of the disclosure is denoted as U . The three membership sets, namely *True*, indeterminacy *Int* and *False* are used to characterize the neutrosophic image I_{NS}. In the image set I^{pre} , a pixel P is denoted as *pix(t, i, f)*. Here, *t*, *i* and *f* point to the values that vary in *True, Int, False*, respectively. Furthermore, from the image domain, *pix(t, i, f)* is converted into the neutrosophic domain. This is shown in Equation (4).

$$I_{NS}(G,H) = \{True(G,H), Int(G,H), False(G,H)\} \tag{4}$$

The notation *True(G, H), Int(G, H), False(G, H)* denotes the membership values, which are mathematically given in Equations (5), (7), and (9), respectively. Here, *d(G, H)* points to the pixel(*G, H*) intensity value and *d(G, H)* is the *d(G, H)*'s local mean value. In addition, δ(*G, H*) is the absolute value corresponding to the difference between the *d(G, H)* intensity and its *d(G, H)* (local mean value), where *z* is a random number generated between 0 and 1.

$$True(G,H) = \frac{\overline{d(G,H)} - \overline{d_{min}}}{\overline{d_{max}} - \overline{d_{min}}} \tag{5}$$

$$\overline{d(G,H)} = \frac{1}{Z \times Z} \sum_{m=G-Z/2}^{G+Z/2} \sum_{n=H-Z/2}^{H+Z/2} d(m,n) \tag{6}$$

$$Int(G,H) = \frac{\delta(G,H) - \delta_{min}}{\delta_{max} - \delta_{min}} \tag{7}$$

$$\delta(G,H) = abs(d(G,H)) - d(G,H) \tag{8}$$

$$False(G,H) = 1 - True(G,H) \tag{9}$$

Geometric mean operation: for $I_{NS}(G,H)$, the indeterminacy is computed by using the value of *Int(G, H)*. In order to correlate with *False, True* with *Int*, the modifications taking place in *True* and *False* should have an effect on the influence of the element and Int's entropy. For the gray space image I^{gray} of I^{pre}, the geometric mean operation is performed rather than the existing α-mean operation.

$$\overline{I^{gray}}(G,H) = \frac{1}{Z \times Z} \prod_{m=G-Z/2}^{G+Z/2} \prod_{n=H-Z/2}^{H+Z/2} (I^{gray}(m,n)) \tag{10}$$

For $I_{NS}()$, the geometric mean operation is computed as per Equations (11)–(18), respectively.

$$\overline{I}_{NS}(\alpha) = I(\overline{True}(\alpha), \overline{Int}(\alpha), False(\alpha)) \tag{11}$$

$$\overline{\overline{T}}rue(\alpha) = \begin{cases} False & Int < (\alpha) \\ \overline{False(\alpha)} & Int \geq (\alpha) \end{cases} \tag{12}$$

$$\overline{False}(\alpha) = \frac{1}{Z \times Z} \sum_{m=G-Z/2}^{G+Z/2} \sum_{n=H-Z/2}^{H+Z/2} True(m,n) \tag{13}$$

$$\overline{False}(\alpha) = \begin{cases} False & Int < (\alpha) \\ \overline{False(\alpha)} & Int \geq (\alpha) \end{cases} \tag{14}$$

$$\overline{False}(\alpha) = \frac{1}{Z \times Z} \sum_{m=G-w/2}^{G+Z/2} \sum_{n=H-w/2}^{H+Z/2} False(m,n) \tag{15}$$

$$\overline{Int}(G,H) = \frac{\overline{\chi_T(G,H)} - \overline{\chi_{T_{min}}}}{\overline{\chi_{T_{max}}} - \overline{\chi_{T_{min}}}} \tag{16}$$

$$\overline{\chi_T}(G,H) = abs(\overline{True}(G,H) - \overline{\overline{True}}(G,H)) \tag{17}$$

$$\overline{\overline{True}}(G,H) = \frac{1}{Z \times Z} \sum_{m=G-Z/2}^{G+w/2} \sum_{n=H-Z/2}^{H+w/2} \overline{True}(m,n) \tag{18}$$

Here, $\overline{\chi_T}(G,H)$ denotes the absolute difference between the $\overline{True}$ (mean intensity) and its means value $\overline{\overline{True}}(i,j)$.

The difference between the mean intensity $\overline{True}$ and its means value $\overline{\overline{True}(i,j)}$ after performing the geometric mean operation is an absolute value $\overline{\delta_E}(G,\ H)$. In this work, α = 0.85. The multi-features (attributes) are extracted from the produced neutrosophic image, which is denoted as I_{NS}. Moreover, to alleviate the segmentation errors, the modified fuzzy C-means (MFCM) segmentation algorithm is introduced in this research work.

The segmentation error can be mathematically given as per Equation (19).

$$Error = \frac{idealobjectpixel - realobjectpixel}{idealobjectpixel} \tag{19}$$

The segmented image acquired from the geometric mean with fuzzy C-means based neutrosophic segmentation is pointed as I^{seg} . Then, from I^{seg} , the multi-features such as ULBP, color features and pixel features are extracted.

3.5. Proposed Feature Extraction Phase: Upgraded LBP, Color Feature and Pixel Feature

The feature extraction is a significant phase in which the most relevant features such as ULBP, color features, and pixel features are extracted. All these extracted features are together used to train the deep learning classifier in the disease detection phase. The extracted features are shown in Figure 3.

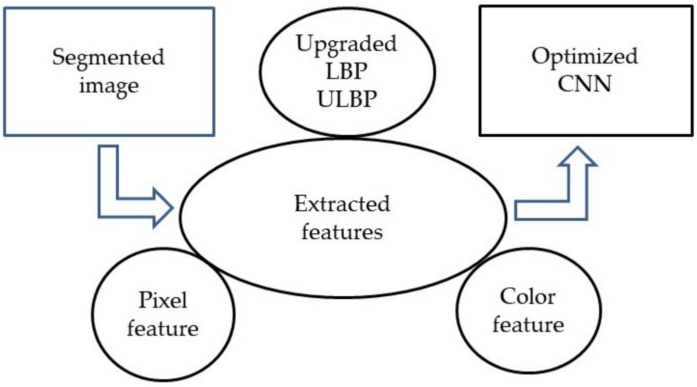

Figure 3. Feature extraction phase.

Upgraded LBP (ULBP)

LBP [23] is a simple and effective texture operator that identifies pixels by thresholding each pixel's neighborhood and interpreting the outcome as a binary integer. The LBP is easier to implement and has a superior discriminative capability. The LBP operator also labels the image pixels along with the decimal numbers.

Each image pixel is calculated with its neighbors during the labeling phase by subtracting the mean pixel value. Additionally, the consequent negative values are encoded as 0. In contrast, the positive and 0 values have been encoded as 1. All of the binary codes are concatenated clockwise from the top-left to obtain a binary number, and these binary numbers are known as LBP codes. The texture descriptor is being used to create the global description, which would be made up of many local descriptions.

Furthermore, the distinguishable capability extracts characteristics from these texture objects. In the block size, 3*3 of I^{seg}, the LBP is applied. The center pixel is referred to as the threshold. The count of neighboring pixels is denoted as pix, and the i^{th} neighboring pixel is denoted as N_{pix}. In addition, P and R denote the neighboring pixel and radius, respectively. Moreover, the intensity of the current and the neighboring pixel is denoted as pix_c and pix_g, respectively. The newly introduced ULBP model is mathematically shown in Equation (20).

$$f^{ULBP}(P,R) = \sum_{i=1}^{Npix} \sigma(\|pix_c - pix_g\|)2^i \tag{20}$$

Here, σ is the standard deviation, which is mathematically given in Equation (21).

$$\sigma = \sqrt{\frac{(X-\mu)^2}{N_{pix}}} \tag{21}$$

Here, μ is the mean of the data point X. Moreover, instead of using the arithmetic mean, we have computed the geometric mean. The extracted ULBP feature is expressed in F^{ULBP} . The value of $\sigma(\|pix_c - pix_g\|)$ is shown in Equation (22).

$$\sigma(\|pix_c - pix_g\|) = \begin{cases} 1 & if(pix_c - pix_g) > 0 \\ 0 & otherwise \end{cases} \tag{22}$$

3.6. Color and Pixel Features

The color features extracted from I^{seg} are the R channel of RGB, H channel of HSV, and L channel of LUV images. The RGB color model is additive [24], wherein "red, green, and blue" light are combined to generate a wide variety of colors. Whereas if an RGB picture is 24-bit, every channel for red, green, and blue contains 8 bits—in other words, the image has been made up of three images (one for each channel), each of which may hold discrete pixels with standard brightness levels of 0 to 255. Each channel of a 48-bit RGB picture (extremely high color depth) has been made up of 16-bit images. The red area of the image appears significantly brighter than the others in the red channel. The HSV (or HSL) color space is a color representation paradigm based on (human) color perception. Color is a term used to describe the hue (mainly red, yellow, green, cyan, blue, or magenta). It is usual to draw a circle around the hue and provide its magnitude in degrees (over 360°). In the "LUV" region (where "L stands for luminance, whereas U and V represent chromaticity values of color images"), the extracted color feature is denoted as f^{color}.

Brightness [25] is a relative term that refers to the intensity of one pixel compared to another. As an image pixel feature, the brightness of each pixel is calculated. The extracted pixel feature is denoted as f^{pixel}. The extracted overall features are denoted as f^{pixel} + f^{color}+f^{ULBP} = F . Using this F, the optimized CNN is trained.

4. Optimal Trained CNN for Disease Detection Model

4.1. Optimized CNN

The extracted F is given as the input to optimized CNN [25–27] for detecting the presence/absence of mango leaf diseases. The CNN comprises three primary layers: "convolutional, pooling, and fully connected layers". For the inputs, the feature representations are learned in the convolutional layer consisting of several convolution kernels. The diverse feature maps are computed using these convolution kernels. In the feature map, every neuron in the current layer is connected to the previous layer's neighboring neurons, and this mechanism is denoted as the neuron's receptive field. Moreover, the input is convoluted by applying the element-wise nonlinear activation function with the learned kernel in order to achieve a new feature map. Several different kernels are being implied to create the complete feature maps. For example, at layer *L* of feature map *K*, the feature value residing in the location (*I*, *J*) can be computed using Equation (23). The generated feature map is denoted as $S^L_{I,J,K}$.

$$S^L_{I,J,K} = W^{L^T}_K \times F^L_{I,J} + Bais^L_K \tag{23}$$

Here, W^L_K and $Bias^L_K$ point to the weight vector corresponding to the feature value residing in the location (*I*, *J*) at layer *L* of feature map *K*. In addition, $F^L_{I,J}$ is the extracted feature, which comes as input to CNN at the location (*I*, *J*) in the L_{th} layer corresponding to the K^{th} map. Moreover, this weight function W is fine-tuned using a new CSUBW model, considering minimizing the loss (error) while detecting (diseased or non-diseased). Within the CNN, the nonlinearity is introduced by the "activation function". The nonlinear activation function is denoted as $AF(.)$. For the convolutional feature $S^L_{I,J,K}$, the activation function $AF^L_{I,J,K}$ is computed using Equation (24).

$$AF^L_{I,J,K} = AF(S^L_{I,J,K}) \tag{24}$$

The typical activation functions are sigmoid, tanh, and ReLU. The shift-invariance of the feature maps is achieved by lessening its resolution. The poling layer is denoted as the pool(). The output from CNN is denoted as $O^L_{I,J,K}$, which can be mathematically given as per Equation (25). Here, $\Re_{i,j}$ points to the local neighborhood that is localized around the location (*I*, *J*). The CNN's final layer is the output layer with more fully connected layers.

$$O^L_{I,J,K} = pool(AF^L_{o,p,K}), \forall(o,p) \in \Re_{I,J} \tag{25}$$

There is *N* count of input–output relations $(F^n, O^n; n \in [1, ..., N]$. Here, *N* points to the n^{th} input data and O^n is the targeted label (presence/absence of disease in mango leaf). The output from CNN is denoted as y^n. In CNN, Equation (26) determines the loss function, which must be minimized. In this research work, the loss function is minimized by fine-tuning the weight of CNN using a newly developed hybrid optimization model

(CSUBW). This is shown in Equation (27). The solution fed as input to the CSUBW model is manifested in Figure 4.

$$Loss = \frac{1}{n}\sum_{n=1}^{N}(\zeta, O^{(n)}, y^{(n)}) \tag{26}$$

$$Obj = min(Loss) \tag{27}$$

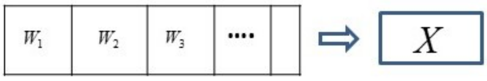

Figure 4. Solution Encoding.

4.2. CNN Training by Proposed CSUBW

The CSO has been developed with the inspiration acquired from the behavior of cats. With higher convergence, the CSO model solves difficult optimization issues. "Seeking mode" and "tracking mode" are the two most common cat behaviors. Furthermore, the BWO was inspired by the "unique mating behavior" of black widow spiders. The BWO model also effectively addresses complex optimization problems with strongly convergent solutions. Furthermore, in this case, the search agents identify global solutions within the search space. According to the literature, hybrid optimization models require a higher level of convergence than standard algorithms [28–34]. We integrated the BWO [35,36] within the CSA model in this study; hence, the proposed hybrid method is the Cat Swarm Updated Black Widow model (CSUBW) model. The stages used in the CSUBW model are shown below:

Step 1: Initialize the search agent's M population (pop) in the D-dimensional space. The velocity of the search agent is denoted, and the position of the search agent is pointed as X.

Step 2: The cats are sprinkled randomly in the dimensional space, and the value is selected randomly within the maximum and minimum velocity bounds.

Step 3: As per the "mixture ratio (MR)", the count of cats is selected and set into the tracing mode. The rest of the cats are set into seeking mode.

Step 4: Seeking Mode

(a) Seeking Mode: for the present cat Cat_k, J count of copies are made. Here, J is the SMP. If SPC value = true, then set J = (SMP-1) and set the present cat as the best one.

(b) As per CDC, the SRD values are randomly plus or minus. Then, replace the old ones with the current ones.

(c) For all the candidate points, the fitness function (Fit) is computed using Equation (27).

$$Pos_i = \frac{Fit_i - Fit_b}{Fit_{max} - Fit_{min}}, where\, 0 < i < j \tag{28}$$

(d) When all are not equivalent, the selecting probability is computed for every candidate point using Equation (28). When the fit is equivalent for every candidate point, the selecting probability is set as 1 for each candidate point.
Here, the objective is minimization, so, $Fit_b = Fit_{max}$

(e) The point is randomly picked to move away from the candidate points, and the position of the cats Cat_k is replaced.

Step 5: Proposed Tracing Mode: In the tracing mode, the cat moves with its own velocity in every dimension. The steps followed in the tracing model are manifested below: (a) For every dimension, the velocity of the search agent is updated using the newly proposed expression given in Equation (29).

$$V_{J_{new}}^{d+1} = \omega V_j^d + \beta(P_g - X_j^d) + \alpha \times \epsilon \tag{29}$$

Here, ω is the inertia weight, and ϵ is the random velocity that is uniformly distributed in the interval [0, 1]. In addition, the controlling parameters are α and β. Mathematically, the control parameter used to control the cats in the exploration process is pointed as $\alpha(t)$ and $\beta(t)$ can be given as per Equations (30) and (31), respectively. Here, $\alpha(min)$ and $\alpha(max)$ point to the minimal and maximal limits. The maximal iteration is denoted as $t(max)$, and the current iteration is pointed as t. Moreover, $\beta(min)$ and $\beta(max)$ denote the value of the first and last iteration, respectively.

$$\alpha(t) = \alpha_{max} - \frac{\alpha_{max} - \alpha_{min}}{t_{max}} \times t \tag{30}$$

$$\beta(t) = \beta_{min} + (\beta_{max} - \beta_{min}) \cdot Sin\frac{\pi t}{t_{max}} \tag{31}$$

(b) Verify whether the velocity resides within the maximum velocity. In case the new velocity is beyond the range of the maximum velocity range, then set it to be equal to the limit.

(c) Update the position of Cat_k using the BWO's mutation update model rather than using the traditional CSA update function. The mutation update model randomly selects the Mutepop number from the population (pop). Based on the mutation rate, the Mutepop is computed.

Step 6: Compute the fitness of the search agent using Equation (27). The cat with the best fitness function is said to be the best solution X_{best}.

Step 7: The cats are moved based on their flags; if the cat Cat_k is found to be in the seeking mode, then apply the seeking mode process; else, apply the tracing mode process.

Step 8: Again, based on the MR, re-pick the count of cats and set them into tracing mode and seeking mode.

Step 9: Terminate.

5. Results and Discussion

5.1. Simulation Setup

The proposed automatic disease detection model was implemented in MATLAB. The proposed work has been evaluated with the data collected from [37]. The sample images acquired after the preprocessing phase are shown in Figure 5. The segmented images acquired after the proposed geometric mean with modified fuzzy C-means based neutrosophic segmentation are shown in Figures 6 and 7. The proposed work has been compared to existing models such as CSO, BWO, WOA, EHO, SVM, NN, NB, and RF. The performance measures such as "Accuracy, Sensitivity, Specificity, Precision, Negative Predictive Value (NPV), F1-Score, False-Positive Rate (FPR), False Negative Rate (FNR), and False Discovery Rate (FDR)" are computed. The proposed model was trained with 70% of the data, and 30% was utilized to test the model. Among these, 70% (considered as 100%), 70%, 80%, and 90% of the training data are adjusted and the results acquired are recorded.

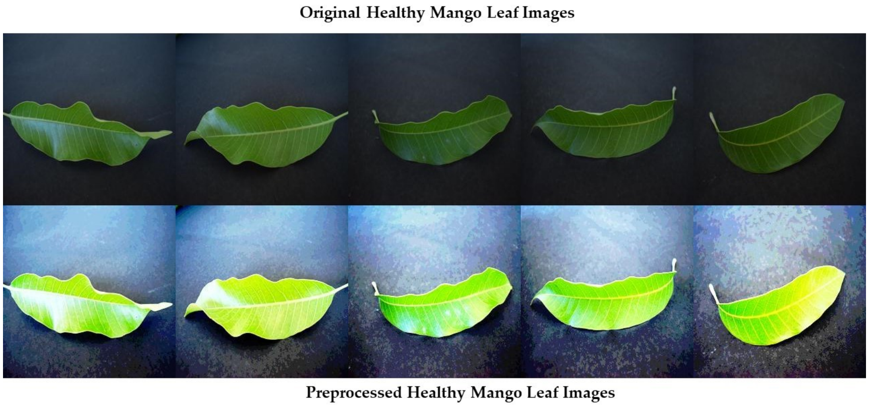
Original Healthy Mango Leaf Images
Preprocessed Healthy Mango Leaf Images

Figure 5. Preprocessing on healthy leaves: original sample image and respective preprocessed image.

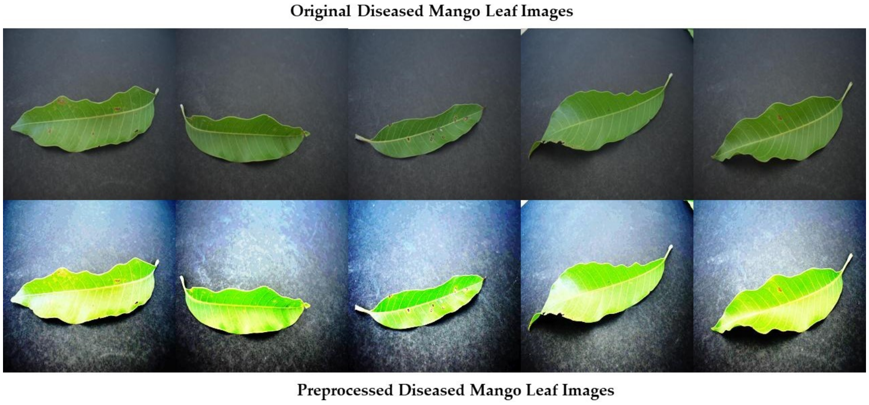
Original Diseased Mango Leaf Images
Preprocessed Diseased Mango Leaf Images

Figure 6. Preprocessing on diseased leaves: original sample image and respective preprocessed image.

5.2. Performance Analysis

The performance of the proposed work (CSUBW + CNN) is compared over the existing molds such as CSO, BWO, WOA, EHO, SVM, NN, NB, and RF, respectively, in terms of positive measures such as “accuracy, Sensitivity, Specificity, Precision” and negative measures such as ”FPR, FNR, FDR, and F1-Score”. In order to prove that the proposed model had achieved the best performance, its positive and the other measures need to be higher; negative measures need to be lower. The proposed work had attained the best performance under all the computed measures. All these improvements are owing to two major reasons: (a) extraction of the most relevant features rather than using the existing ones and (b) fine-tuning the parameter of the detection framework (CNN) via the newly introduced hybrid optimization model. In the newly introduced optimization model, we have considered four major parameters: ω is the inertia weight, ϵ is the random velocity that is uniformly distributed in the interval [0, 1], and α and β are the controlling parameters. All these together aided in boosting the convergence performance of the projected model. The positive performance of the CSUBW + CNN is manifested in Figure 8. These evaluations vary the learning rate from 70% to 80%, and 90%, respectively. The CSUBW + CNN had attained maximal accuracy under all the three variations in the learning rate on observing the outcomes. When the LR = 07, the CSUBW + CNN had attained the maximal accuracy as 93%, while the existing models had recorded the lower accuracy ranges as CSO = 0.65, BWO = 0.68, WOA = 0.69, EHO = 0.7, SVM = 0.6, NN = 0.4, NB = 0.7 and RF = 0.62. Moreover, the specificity, sensitivity, and precision of the CSUBW + CNN is also higher under all the variations in the LR. At LR = 90, the precision of the CSUBW + CNN has achieved the maximal value of 100%, which is the most optimal score. The sensitivity of the CSUBW + CNN at LR = 90 is 100%, which is also the best score. On the other hand, the FDR, FNR, FDR and F1-score’s performance of the proposed model are shown in Figure 9. The FDR of the CSUBW + CNN had attained the least value below 0.06 for every variation in the LR. At the 90th LR, the CSUBW + CNN had attained the least FDR as 0.04, which is 90%, 90.2%, 90.4%, 92.4%, 88.5%, 60%, 89.4%, and 92% better than the existing models such as CSO, BWO, WOA, EHO, SVM, NN, NB, and RF, respectively. The FPR of the CSUBW + CNN had attained the least value for every variation in the LR. In addition, F1-scores are computed to validate the efficiency of the CSUBW + CNN. On observing the outcomes from Figure 9, the CSUBW + CNN had attained the maximal value for every variation in the LR. The F1-score of the CSUBW + CNN at LR = 70 is 92%, which is better than the existing models such as CSO = 70%, BWO = 71%, WOA = 71.5%, EHO = 72%, SVM = 75%, NN = 10%, NB = 72% and RF = 70%. Thus, from the evolutionist, it is vivid that the proposed work had attained the most favorable performance, and hence, it is sufficient for detecting the mango leaf disease.

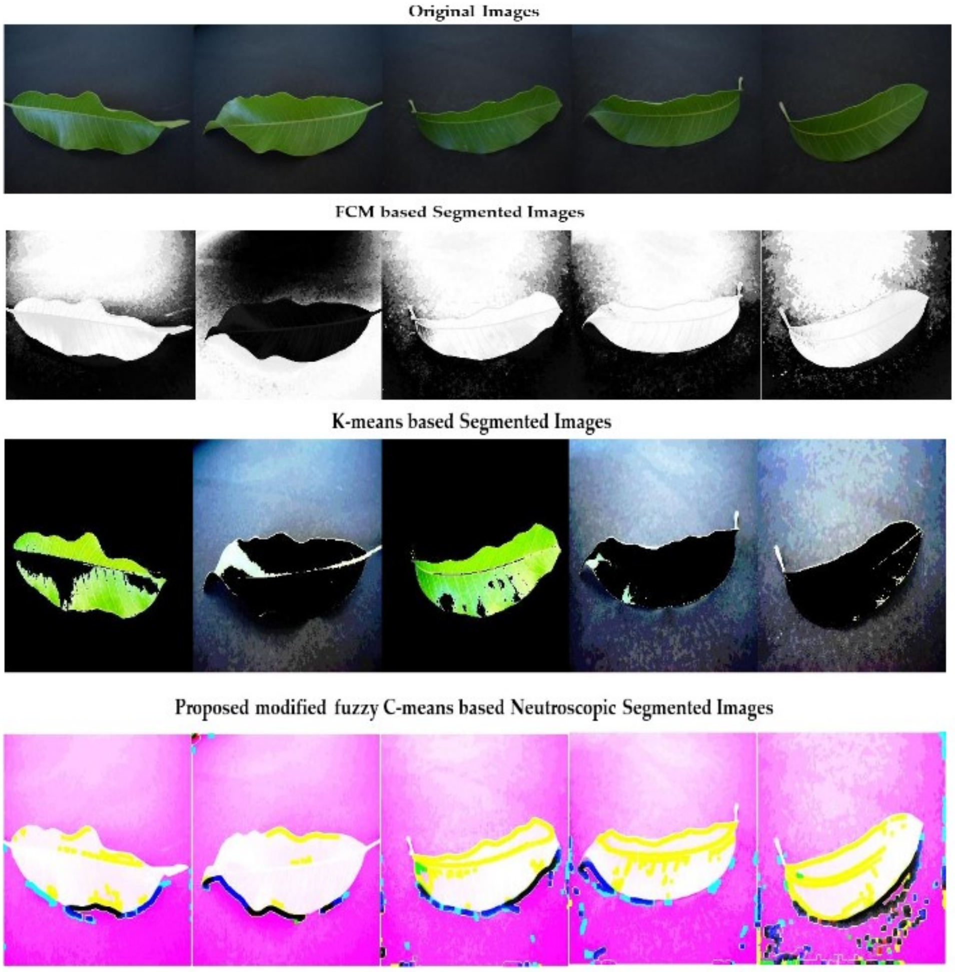
Original Images
FCM based Segmented Images
K-means based Segmented Images
Proposed modified fuzzy C-means based Neutroscopic Segmented Images

Figure 7. Figure showing the difference between proposed segmentation approach with FCM-based segmented image and K-means-based segmented image.

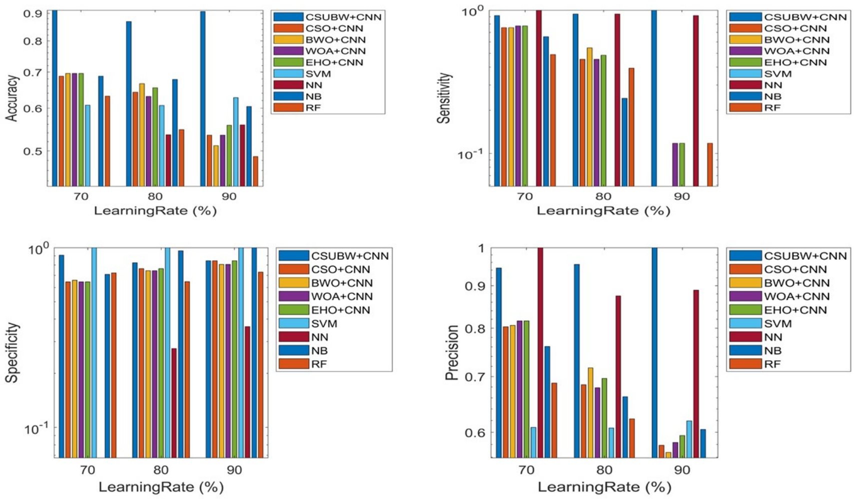

Figure 8. Performance comparison of proposed and conventional models: Accuracy, Sensitivity, Specificity, and Precision.

5.3. Convergence Analysis by Fixing ω = 0.2, 0.5 and 0.8

The convergence analysis is undergone to prove that the proposed algorithm is more significant than the existing algorithms, particularly in achieving the defined objective function. Within the proposed optimization model, two parameters (ω is the inertia weight, and ϵ is the random velocity uniformly distributed with the interval [0, 1] in Equation (29)) have been introduced. So, we have varied ω from 0.2 to 0.5 and 0.8, respectively, and ϵ from 0.2 to 0.5 and 0.8, respectively. The results acquired are manifested in Figure 4. While fixing ω = 0.2, the cost functions of the CSUBW with variation in ϵ = 0.2, 0.5 and 0.8 are manifested in Figure 10. Since this work aims to lessen the loss function of CNN (the leaf disease detector), the parameter that archives the least value is said to be the best one. On observing the outcome, the cost function of the CSUBW at the 2nd iteration is higher for all the three variations (i.e., ϵ = 0.2, ϵ = 0.5 and ϵ = 0.8). However, as the count of iteration tends to increase, the cost function of ϵ = 0.2, ϵ = 0.5 and ϵ = 0.8 also minimized. Inherently, among all the three variations (ϵ = 0.2, ϵ = 0.5 and ϵ = 0.8), the least value has been recorded by the CSUBW at ϵ = 0.2. At ϵ = 0.2, the cost function of the CSUBW at 25th iteration is 2.31% and 3.21% better than the cost function of the CSUBW at ϵ = 0.2 and ϵ = 0.5, respectively. Then, on fixing ω= 0.5 for the CSUBW, the variations taking place under ϵ = 0.2, ϵ = 0.5 and ϵ = 0.8 are recorded in Figure 11. On observing the outcomes, the CSUBW had attained the least value, when ϵ = 0.2, and the attained cost function at the 25th iteration is 1.0667 (best score). In addition, the value of ω = 0.8 is set for the CSUBW, and the value of ϵ is varied from ϵ = 0.2, ϵ = 0.5 and ϵ = 0.8, respectively. The results acquired by the proposed wok are manifested in Figure 12. Under this scenario, the CSUBW had attained the least cost function (i.e., achievement of the objective function) when ϵ = 0.2. Furthermore, at ω = 0.8, the cost function of the CSUBW at the 25th iteration is 0.93% and 2.30% better than the cost function of the CSUBW at ϵ = 0.2 and ϵ = 0.5, respectively.

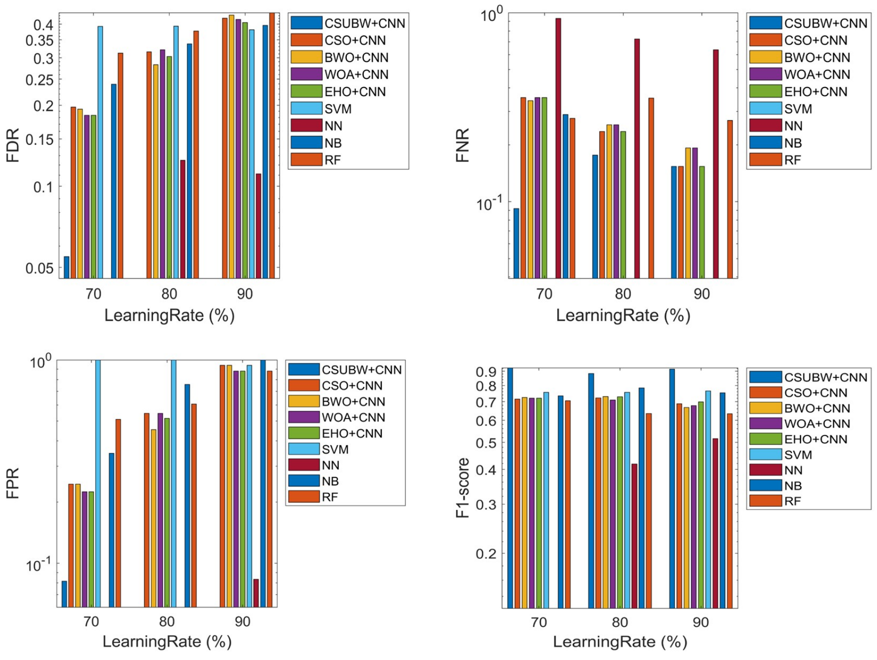

Figure 9. FDR, FNR, FPR and F1-score-based performance comparison of proposed and conventional models.

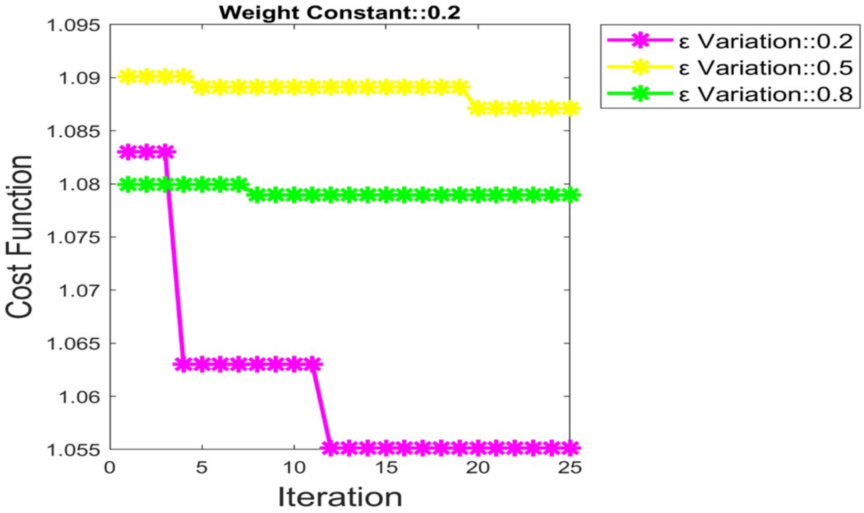

Figure 10. Cost function of the CSUBW by fixing ω = 0.2, 0.5, 0.8; varying ϵ = 0.2.

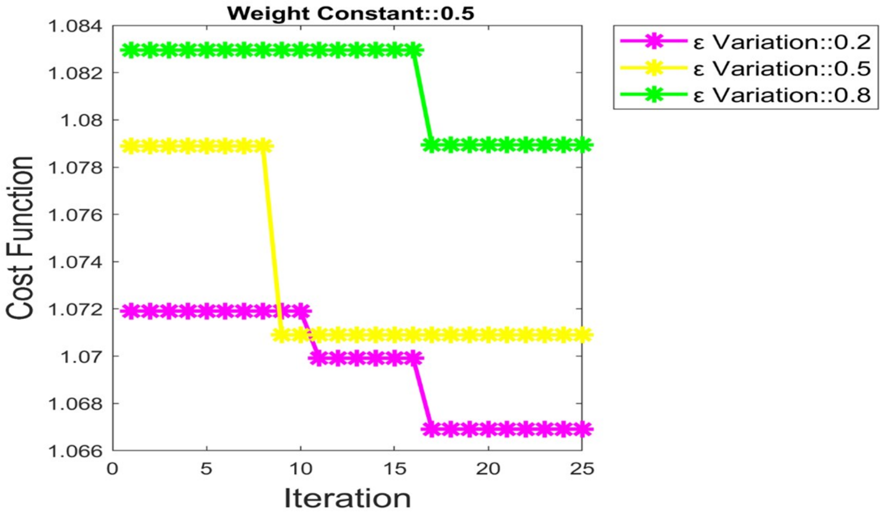

Figure 11. Cost function of the CSUBW by fixing ω = 0.2, 0.5, 0.8; varying ϵ = 0.5.

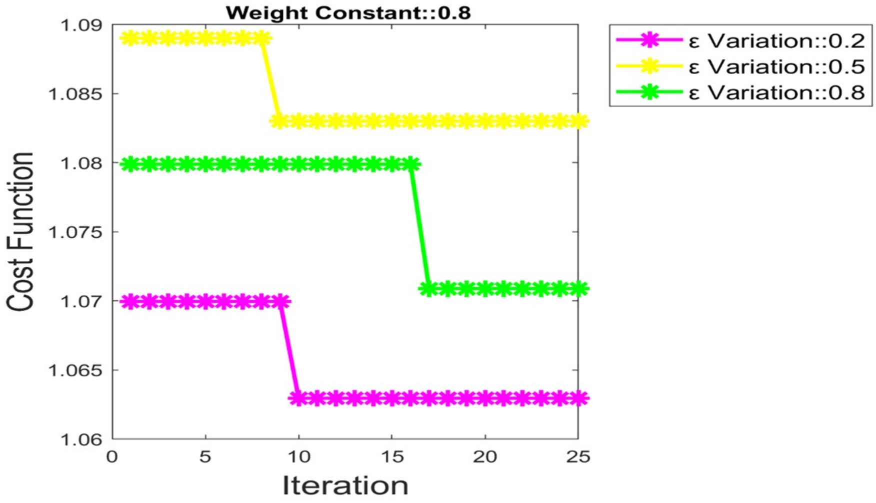

Figure 12. Cost function of the CSUBW by fixing ω = 0.2, 0.5, 0.8; varying ϵ = 0.8.

5.4. Convergence Analysis by Fixing ϵ = 0.2, 0.5 and 0.8

The convergence of the CSUBW is evaluated by fixing ϵ = 0.2, 0.5 and 0.8 and by varying ω from 0.2 to 0.5 and 0.8, respectively. The results acquired with the CSUBW for fixing ϵ = 0.2 are manifested in Figure 13. On observing the outcome acquired by fixing ϵ = 0.2, the CSUBW has achieved the objective function (minimized cost function) at ω = 0.2. At the 25th iteration, the cost function of the CSUBW is 0.93% and 1.04% better than the cost function of the CSUBW at ϵ = 0.5 and ϵ = 0.8, respectively. In addition, results acquired with the CSUBW for fixing ϵ = 0.5 are manifested in Figure 14. On observing the outcomes, the CSUBW had attained the best score at ω = 0.8, and it is 1.1% and 1.65% better than the cost function of the CSUBW at ϵ = 0.2 and ϵ = 0.5, respectively. Similar to this, the convergence analysis of the CSUBW for ϵ = 0.8 is recorded is Figure 15. For the CSUBW, the value of ϵ as well as ω are varied in Equation (29). When ω = 0.2 and ϵ = 0.2, the CSUBW had attained the optimal value as 1.0551, which is the most favorable score. In addition, when ω = 0.5 and ϵ = 0.5, the CSUBW had attained the best score as 1.0709. Under the case, ω = 0.8 and ϵ = 0.8, the CSUBW had attained the optimal performance value as 1.0709.

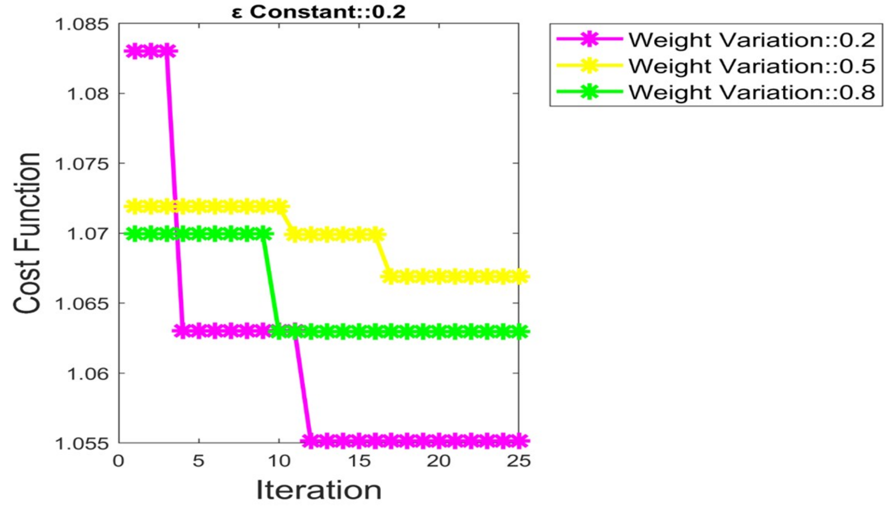

Figure 13. Cost function of the CSUBW by fixing ϵ = 0.2, 0.5 and 0.8; varying ω = 0.2.

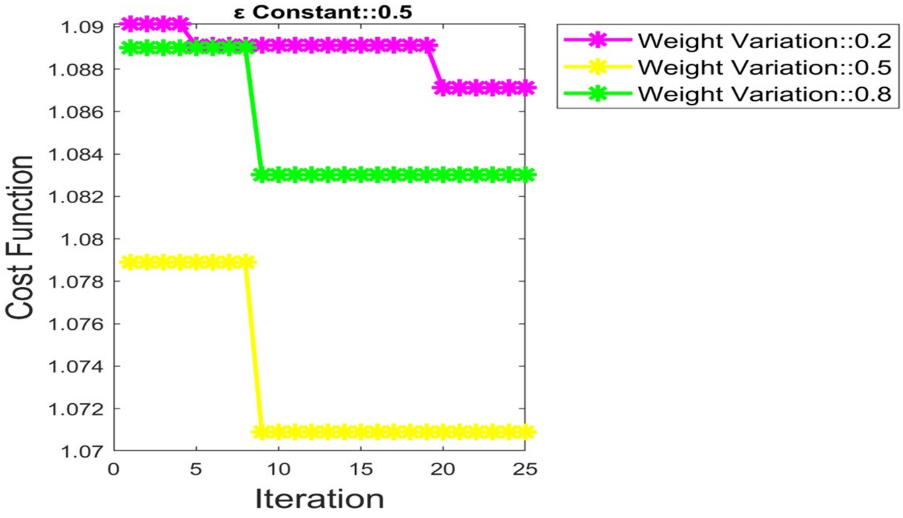

Figure 14. Cost function of the CSUBW by fixing ϵ = 0.2, 0.5 and 0.8; varying ω = 0.5.

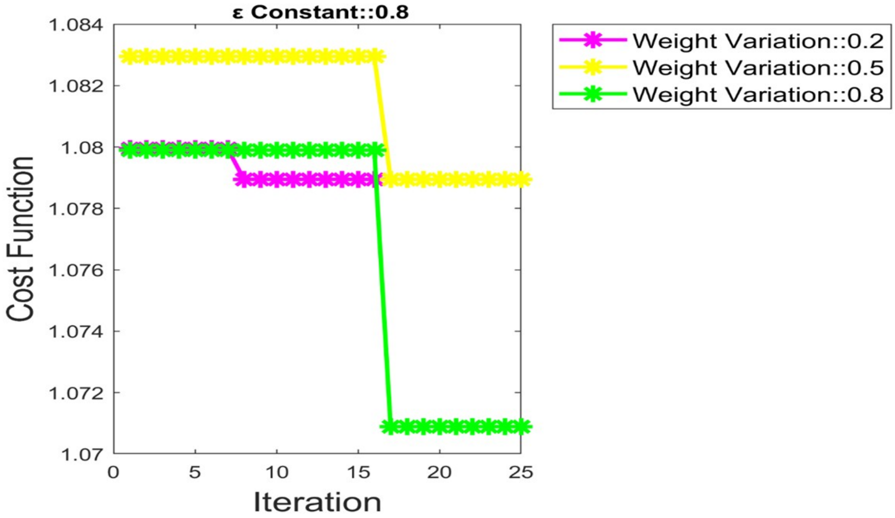

Figure 15. Cost function of the CSUBW by fixing ϵ = 0.2, 0.5 and 0.8; varying ω = 0.8.

5.5. Overall Performance Analysis

The overall performance of the CSUBW + CNN had been recorded for both the proposed and existing models at 70% of LR. The projected model had attained the best results, owing to the enhancement of the convergence speed of the solutions with the newly introduced hybrid optimization model. The accuracy of the CSUBW + CNN is 0.912, which is 24.5%, 23.6%, 23.6%, 23.6%, 33.3%, 52.8%, 24.56% and 30.7% better than the existing models such as CSO, BWO, WOA, EHO, SVM, NN, NB, and RF, respectively. In addition, the specificity of the CSUBW + CNN is 28.9%, 27.5%, 28.9%, 28.9%, 10.1%, 92.5%,21.7% and 20.2% better than the existing models such as CSO, BWO, WOA, EHO, SVM, NN, NB and RF, respectively. The CSUBW + CNN had attained the maximal precision of 0.94521. On the other hand, the negative measures such as FPR, FDR, and FNR of the CSUBW + CNN had attained the minimal value (best score). The FNR of the CSUBW + CNN is 0.092105, which is better than CSO = 0.35526, BWO = 0.34211, WOA = 0.35526, EHO = 0.35526, NN = 0.93243, NB = 0.28947 and RF = 0.27632.

In addition, the CSUBW + CNN had attained the maximal performance in terms of other measures such as F1-score. The CSUBW + CNN had attained the maximal F1-score as 0.92617, which is 22.7%, 21.7%, 22.17%, 22.19%, 18.35%, 86.3%, 20.6% and 23.86% better than the existing models such as CSO, BWO, WOA, EHO, SVM, NN, NB, and RF, respectively. Thus, from the overall evaluation, it is clear that the CSUBW + CNN had maximal performance and hence becomes much sufficient for mango leaf disease detection. Figures 16–18 depict the ROC curve for varied learning percentages 70, 80 and 90. The ROC curve is considered for the predicted label and the original label. The proposed methodology is compared over the NB, RF, NN, SVM, EHO + CNN, WOA + CNN, BWO + CNN and CSO + CNN. The value of the proposed method is high when compared to the other existing models.

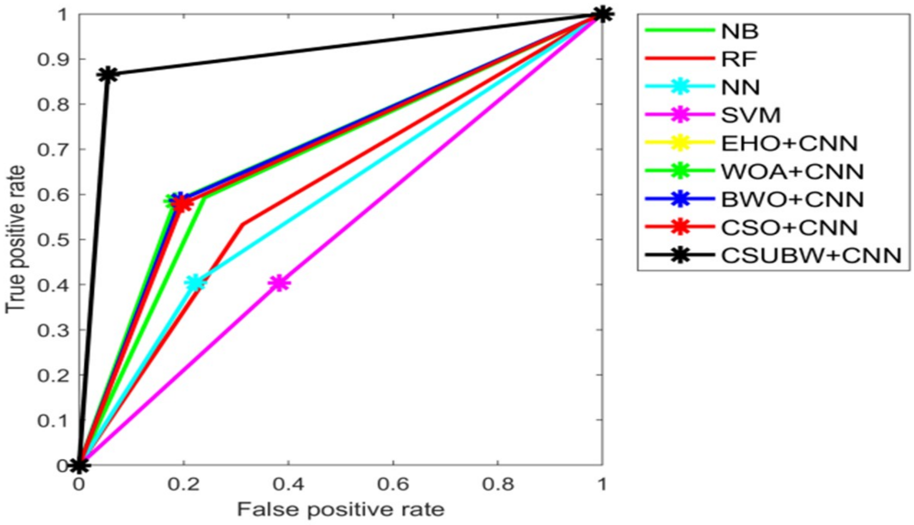

Figure 16. ROC Curve for 70th Learning Percentage.

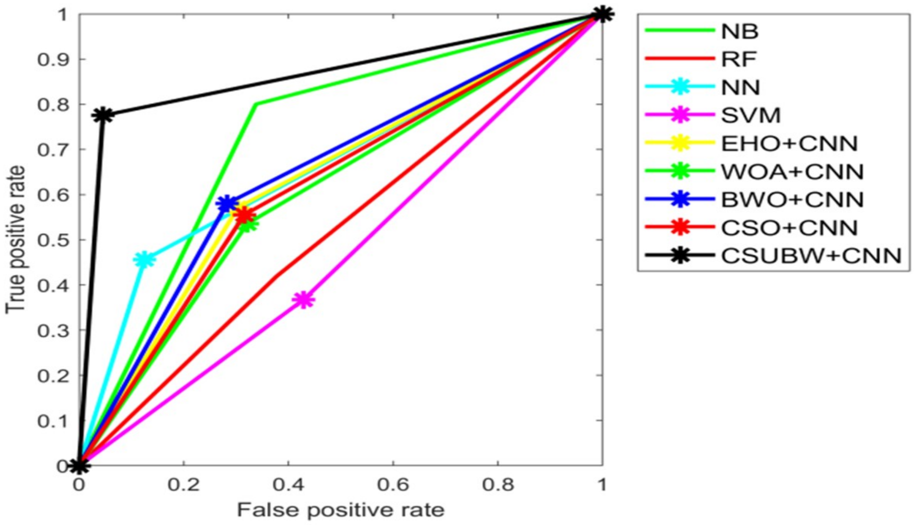

Figure 17. ROC Curve for 80th Learning Percentage.

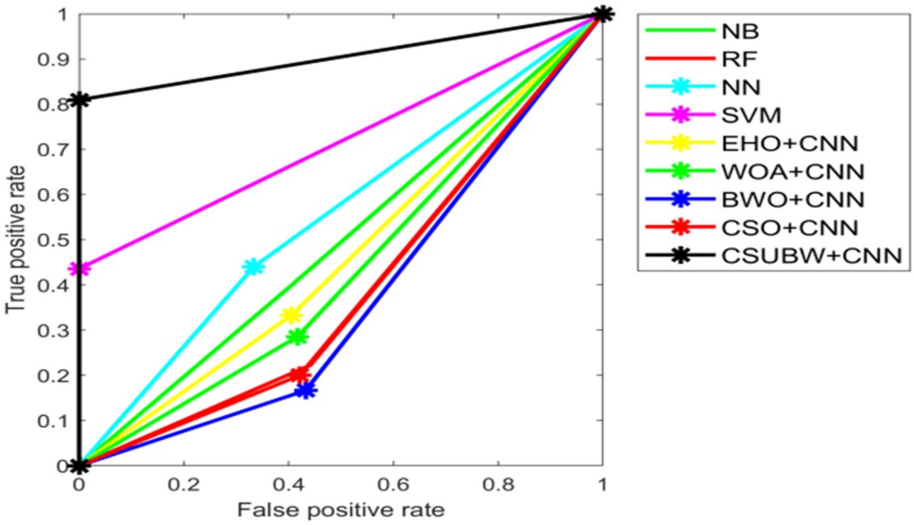

Figure 18. ROC Curve for 90th Learning Percentage.

6. Conclusions and Future Scope

This paper proposed a new automatic disease prediction model by following four major phases: (a) preprocessing, (b) image segmentation, (c) feature extraction, and (d) disease prediction. The acquired raw image was first preprocessed using contrast enhancement and histogram equalization to eliminate noise and other undesirable artifacts and improve the image quality. Then, preprocessed images were segmented via Geometric Mean-based neutrosophic with fuzzy C-mean. Next, the most important features were retrieved from the segmented pictures, including "texture features such as ULBP, color features, and pixel features". Finally, these characteristics were given into the detection phase based on a CNN model for illness detection. Furthermore, to enhance the classification accuracy of CNN, its weights were fine-tuned using the CSUBW model. Finally, the new hybrid optimization model (CSUBW) hybridized the standard CSO and BWO algorithms. The performance of the proposed work had been recorded for both the proposed and existing models at 70% of LR. The accuracy of the proposed work is 0.912, which is 24.5%, 23.6%, 23.6%, 23.6%, 33.3%, 52.8%, 24.56% and 30.7% better than the existing models such as CSO, BWO, WOA, EHO, SVM, NN, NB, and RF, respectively. Thus, from the overall evaluation, it is clear that the proposed work had attained the maximal performance and hence is sufficient for mango leaf disease detection. The current research work has emphasized assisting the growers by helping them identify the mango leaves disease in their environment. However, we intend to better support them by developing enhanced feature extraction methods and detection via a most effective optimization approach.

Author Contributions: M.M. and A.K.P. prepared the initial draft and performed the experiments; P.K.M. performed the visualization; A.K.P., S.K., M.Z. supervised the research; S.K. proofread and prepared the final manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All data has been presented in the main text.

Acknowledgments: The authors would like to acknowledge the support of the Ministry of Science and Higher Education of the Russian Federation (Government Order FENU-2020-0022).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Chouhan, S.S.; Singh, U.P.; Jain, S. Web Facilitated Anthracnose Disease Segmentation from the Leaf of Mango Tree Using Radial Basis Function (RBF) Neural Network. *Wirel. Pers. Commun.* **2020**, *113*, 1279–1296. [CrossRef]
2. Mia, M.; Roy, S.; Das, S.K.; Rahman, M. Mango leaf disease recognition using neural network and support vector machine. *Iran. J. Comput. Sci.* **2020**, *3*, 185–193. [CrossRef]
3. Nagaraju, Y.; Sahana, T.S.; Swetha, S.; Hegde, S.U. Transfer Learning based Convolutional Neural Network Model for Classification of Mango Leaves Infected by Anthracnose. In Proceedings of the IEEE International Conference for Innovation in Technology (INOCON), Bangluru, India, 6–8 November 2020; pp. 1–7. [CrossRef]
4. Sujatha, S.; Saravanan, N.; Sona, R. Disease identification in mango leaf using image processing. *Adv. Nat. Appl. Sci.* **2017**, *11*, 1–8.
5. Singh, U.P.; Chouhan, S.S.; Jain, S.; Jain, S. Multilayer Convolution Neural Network for the Classification of Mango Leaves Infected by Anthracnose Disease. *IEEE Access* **2019**, *7*, 43721–43729. [CrossRef]
6. Kabir, R.; Jahan, S.; Islam, M.R.; Rahman, N.; Islam, M.R. Discriminant Feature Extraction using Disease Segmentation for Automatic Leaf Disease Diagnosis. In Proceedings of the International Conference on Computing Advancements (ICCA 2020), Dhaka, Bangladesh, 10–12 January 2020.
7. Liu, L.; Li, J.; Sun, Y. Research on the Plant Leaf Disease Region Extraction. In Proceedings of the International Conference on Video, Signal and Image Processing (VSIP 2019), Wuhan, China, 29–31 October 2019.
8. Su, T.; Mu, S.; Dong, M.; Sun, W.; Shi, A. nAn Improved TrAdaBoost for Image Recognition of Unbalanced Plant Leaf Disease. In Proceedings of the 2019 8th International Conference on Computing and Pattern Recognition, Beijing, China, 23–25 October 2020.

9. Trongtorkid, C.; Pramokchon, P. Expert system for diagnosis mango diseases using leaf symptoms analysis. In Proceedings of the 2018 International Conference on Digital Arts, Media and Technology (ICDAMT), Phayao, Thailand, 25–28 February 2018; pp. 59–64.
10. Trang, K.; TonThat, L.; Thao, N.G.M.; Thi, N.T.T. Mango Diseases Identification by a Deep Residual Network with Contrast Enhancement and Transfer Learning. In Proceedings of the 2019 IEEE Conference on Sustainable Utilization and Development in Engineering and Technologies (CSUDET), Penang, Malaysia, 7–9 November 2019; pp. 138–142. [CrossRef]
11. Tumang, G.S. Pests and Diseases Identification in Mango using MATLAB. In Proceedings of the 2019 5th International Conference on Engineering, Applied Sciences and Technology (ICEAST), Luang Prabang, Laos, 2–5 July 2019; pp. 1–4. [CrossRef]
12. Madiwalar, S.C.; Wyawahare, M.V. Plant disease identification: A comparative study. In Proceedings of the 2017 International Conference on Data Management, Analytics and Innovation (ICDMAI), Pune, India, 24–26 February 2017; pp. 13–18. [CrossRef]
13. Arya, S.; Singh, R. A Comparative Study of CNN and AlexNet for Detection of Disease in Potato and Mango leaf. In Proceedings of the 2019 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT), Ghaziabad, India, 27–28 September 2019; pp. 1–6. [CrossRef]
14. Setyanto, A.; Agastya, I.M.A.; Priantoro, H.; Chandramouli, K. User Evaluation of Mobile based Smart Mango Pest Identification. In Proceedings of the 2020 8th International Conference on Cyber and IT Service Management (CITSM), Pangkal, Indonesia, 23–24 October 2020; pp. 1–7. [CrossRef]
15. Swetha, K.; Venkataraman, V.; Sadhana, G.D.; Priyatharshini, R. Hybrid approach for anthracnose detection using intensity and size features. In Proceedings of the 2016 IEEE Technological Innovations in ICT for Agriculture and Rural Development (TIAR), Chennai, India, 15–16 July 2016; pp. 28–32. [CrossRef]
16. Pham, T.N.; Van Tran, L.; Dao, S.V.T. Early Disease Classification of Mango Leaves Using Feed-Forward Neural Network and Hybrid Metaheuristic Feature Selection. *IEEE Access* **2020**, *8*, 189960–189973. [CrossRef]
17. Nishat, M.M.; Faisal, F. An Investigation of Spectroscopic Characterization on Biological Tissue. In Proceedings of the 2018 4th International Conference on Electrical Engineering and Information & Communication Technology (iCEEiCT), Dhaka, Bangladesh, 13–15 September 2018; pp. 290–295. [CrossRef]
18. Yamparala, R.; Challa, R.; Kantharao, V.; Krishna, P.S.R. Computerized Classification of Fruits using Convolution Neural Network. In Proceedings of the 2020 7th International Conference on Smart Structures and Systems (ICSSS), Chennai, India, 23–24 July 2020; pp. 1–4. [CrossRef]
19. Ullagaddi, S.B.; Raju, S.V. Disease recognition in Mango crop using modified rotational kernel transform features. In Proceedings of the 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 6–7 January 2017; pp. 1–8. [CrossRef]
20. Kumar, P.; Ashtekar, S.; Jayakrishna, S.S.; Bharath, K.P.; Vanathi, P.T.; Kumar, M.R. Classification of Mango Leaves Infected by Fungal Disease Anthracnose Using Deep Learning. In Proceedings of the International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 8–10 April 2021. [CrossRef]
21. Kaur, S. Contrast Enhancement Techniques for Images—A Visual Analysis. *Int. J. Comput. Appl.* **2013**, *64*, 20–25.
22. Histogram Equalization. Available online: https://towardsdatascience.com/histogram-equalization-5d1013626e64 (accessed on 4 August 2021).
23. Sairamya, N.J.; Susmitha, L.; George, S.T.; Subathra, M.S.P. Hybrid approach for classification of electroencephalographic signals using time–frequency images with wavelets and texture features. In *Intelligent Data Analysis for Biomedical Applications*; Academic Press: Cambridge, MA, USA, 2019; Chapter 12.
24. Pixel Feature. Available online: https://www.tutorialspoint.com/dip/brightness_and_contrast.htm (accessed on 4 August 2021).
25. Chandanapalli, S.B.; Reddy, E.S.; Lakshmi, D.R. Convolutional Neural Network for Water Quality Prediction in WSN. *J. Netw. Commun. Syst.* **2019**, *2*, 40–47.
26. Gangappa, M.; Mai, C.; Sammulal, P. Enhanced Crow Search Optimization Algorithm and Hybrid NN-CNN Classifiers for Classification of Land Cover Images. *Multimed. Res.* **2019**, *2*, 12–22.
27. Darekar, R.V.; Dhande, A.P. Emotion Recognition from Speech Signals Using DCNN with Hybrid GA-GWO Algorithm. *Multimed. Res.* **2019**, *2*, 12–22.
28. Beno, M.M.; Valarmathi, I.R.; Swamy, S.M.; Rajakumar, B.R. Threshold prediction for segmenting tumour from brain MRI scans. *Int. J. Imaging Syst. Technol.* **2014**, *24*, 129–137. [CrossRef]
29. Wagh, M.B.; Gomathi, N. Improved GWO-CS Algorithm-Based Optimal Routing Strategy in VANET. *J. Netw. Commun. Syst.* **2019**, *2*, 34–42.
30. Srinivas, V.; Santhirani, C. Hybrid Particle Swarm Optimization-Deep Neural Network Model for Speaker Recognition. *Multimed. Res.* **2020**, *3*, 1–10.
31. Netaji, V.K.; Bhole, G.P. Optimal Container Resource Allocation Using Hybrid SA-MFO Algorithm in Cloud Architecture. *Multimed. Res.* **2020**, *3*, 11–20.
32. Ashok Kumar, C.; Vimala, R. Load Balancing in Cloud Environment Exploiting Hybridization of Chicken Swarm and Enhanced Raven Roosting Optimization Algorithm. *Multimed. Res.* **2020**, *3*, 45–55.
33. Roy, M.R.G. Economic dispatch problem in power system using hybrid Particle Swarm optimization and enhanced Bat optimization algorithm. *J. Comput. Mech. Power Syst. Control* **2020**, *3*, 27–33.

34. Al Raisi, A.A.J. Hybird Particle Swarm Optimization and Gravitational Search Algorithm for economic dispatch in power system. *J. Comput. Mech. Power Syst. Control* **2020**, *3*, 34–40. [CrossRef]
35. Chu, S.C.; Tsai, P.W.; Pan, J.S. Cat Swarm Optimization. In Proceedings of the 9th Pacific Rim International Conference on Artificial Intelligence, Guilin, China, 7–11 August 2006; pp. 854–858.
36. Hayyolalam, V.; Kazem, A.A.P. Black Widow Optimization Algorithm: A novel met. a-heuristic approach for solving engineering optimization problems. *Eng. Appl. Artif. Intell.* **2020**, *87*, 103249. [CrossRef]
37. Available online: https://www.kaggle.com/agbajeabdullateef/a-database-of-leaf-images-from-mendeley-data (accessed on 1 August 2021).