Feature Extraction and Classification Using Linear Principle Component Analysis and Recurrent Neural Network in Banana Stem Disease Detection

T. Mahendran

Assistant Professor, Department of Computer Applications,
Arignar Anna Government Arts College, Villupuram – 605 602. Tamilnadu. India.
Email: maheinfex@gmail.com

Abstract

Agriculture is the foundation of each country on the planet. In India, the greater part of the provincial populace actually relies upon horticulture. The rural area gives significant work in rustic zones. The infections on the banana are a significant issue which makes the sharp decline in the creation of banana. These require cautious determination and ideal dealing with to shield the harvests from weighty misfortunes. Presently a day's yield faces numerous illnesses. The unaided eye perception of specialists is the principle approach embraced by and by for location and distinguishing proof of banana stem illnesses. In any case, this necessities persistent observing of specialists which may be restrictively costly in enormous homesteads. So this paper proposes the classification and feature extraction in detecting the banana stem diseases. When the disease is detected in stem of the plant, it will help the experts in carrying out the prevention methods. Here the enhanced image processing and machine learning techniques are adopted for enhancing the classification accuracy in disease detection. So we use linear principle component analysis (L-PCA) and recurrent neural network (RNN) for extraction and classifying the input images from dataset. We initially pre-process and segment the image, where the segmentation is done using average adaptive thresholding which segment the diseased part based on the threshold value of the image. The simulation result shows the enhanced accuracy, precision and recall.

Keywords: banana stem diseases, L-PCA, RNN, average adaptive thresholding, accuracy, precision and recall.

1. INTRODUCTION

Agriculture efficiency is something on which economy profoundly depends. This is the one reason that illness discovery in plants assumes a significant part in agribusiness field, as having sickness in plants are very characteristic. Programmed plant sickness distinguishing proof and order is likewise an unavoidable territory that requires the advancement of robotized PC vision or machine vision framework with the utilization of picture handling procedure. Significant banana sicknesses that express the manifestations on leaves are

panama illness, moko infection, sigatoka illness, dark spot, banana bunchy top, irresistible chlorosis, banana streak infection and banana bract mosaic infection sickness [1].

The administration of harvests requires close checking particularly for the distinguishing proof of infections that can influence creation and economy of a country. The Machine Learning can be utilized to distinguish the sicknesses in harvests. There are numerous things that can make various infections the plants, which prompts harms in rural field and totally the economy of the country. On the off chance that plant infections are identified successfully and overseen precisely, at that point we can evades the misfortunes in harvest fields. So to make the farming fields and the economy of the nation more grounded, fast and precise sickness recognition framework is required [2]. Banana (Musa spp) is a crucial food in numerous nations of the muggy jungles. It is among the main starch sources in the eating regimen of individuals in these locales. It needs low work prerequisite and generally high-energy yield make plantain a reasonable fundamental for regions where work deficiency is normally the primary imperative to creation. Regardless of the way that harvest is filled in different agro climatic states of the country, its creation is consistently being obstructed by various sicknesses exacting yield misfortunes of amazing measurement both in amount just as quality perspectives. The banana development might be influenced by certain understood elements like Nutrients lop-sidedness and Bacterial or parasitic assaults and express factors like floods, dry spells and so on The banana leaves are influenced by numerous illnesses, for example, Black sigatoka, Panama shrink and Mosaic infection. In Black sigatoka, a light yellow/dull or earthy colored/dark streaks are available corresponding to the vein of the leaf. These streaks expand to shape trademark spots having dim caramel to dark tone, direct oval regions with not well characterized line, frequently encompassed by yellow radiance [3].

New method in picture handling and item identification with more prominent exactness is Deep learning in the grouping of different harvest sicknesses. Move training is single such mainstream technique in profound realizing, where pre-prepared prototypes are adjusted to do another errand.

Profound exchange learning creates a new structure for advanced picture handling and prescient examination, with more prominent precision and has tremendous potential in harvest illness identification. DTL method additionally provides a capable road for in-field infection acknowledgment utilizing enormous prepared picture datasets and offers an alternate route to the created prototypes to encounter the limitations that are presented through versatile purpose. This would have an unmistakable down to earth an incentive for genuine field climate. Prior examinations have approved AI-based acknowledgment of harvest illnesses in wheat, cassava and on datasets of sound and infected plants. Harvest infection acknowledgement dependent on a mechanized picture framework through component extraction has uncovered promising outcomes [4] however extricating highlights is computationally thorough and includes master information for vigorous portrayal.

Just barely any limited huge, curated picture datasets of harvest infection library exist. The Plant Village stage holds more than 50,000 pictures of various yields and illnesses. In any

case, the greater part of these pictures was taken with segregated plants on a plain foundation, and CNN prepared on these pictures didn't accomplish well when utilizing genuine field pictures. To assemble powerful and more common sense identification models, a lot of solid and unhealthy pictures taken from various tainted pieces of the plants, and developing under various ecological conditions are required. These pictures consequently should be marked and pre-screened by plant pathology specialists. Up until this point, existing yield sickness location models are generally zeroing in on leaf indications. Sadly, various side effects additionally show up in different pieces of the herb then the pre-eminent models are banana irritation and sickness connected indications [5]. The remainder of the paper is coordinated as follows: in Section 2, we present the connected work. Area 3 clarifies specialized subtleties of the proposed approach and the design. Besides, exploratory assessment and results are accounted for in Section 4. At last, Section 5 closes the paper and gives a standpoint to future work.

2. RELATED WORKS

Plant illness discovery had been a significant worry for scientists and the space of picture preparing and AI had gigantic commitment towards this reason. A programmed discovery technique is projected by Sandip. P Bhamareatel [6] for Black Sigatoka infection in Banana vegetation. At this point the creator is securing the picture physically as well as is utilizing foundation deduction strategy for object extraction and utilizing provincial and neighbourhood include extraction technique for distinguishing the banana vegetation influenced with Black Sigatoka. A different product answer for programmed banana vegetation infection location is suggested through Basavaraj Tigadietal [7]. The shading highlights and histogram of layout highlights were separated after the pictures of leaf.

The removed highlights are taken care of into the Artificial Neural Network (ANN) for characterization of the sound and irregular leaves. Jihen Amaraatel. [8] Suggested a framework based CNNs for perceiving the banana vegetation and for characterization of the banana vegetation illnesses. An insignificant picture handling based LeNet design [9] is suggested. At this point, the creator guarantees his framework to absorb visual highlights straightforwardly by looking at the pictures of influenced vegetation leaves with the great ones. An exactness of 98% is accomplished by the suggested framework. Godliver Owomugishaatel. [10] The suggested a PYTHON stage on basis of framework for distinguishing banana vegetation infections utilizing k-Fold cross-approval technique. Here various classification techniques are looked at for identifying banana vegetation sickness. P. Geurts et al. [11] tree classification techniques execution to be the awesome distinguishing banana bacterial shrink and banana Black Sigatoka illnesses. Samajpati, B. J [12] utilized surface acknowledgment strategy for identification of apple natural product infection like apple smear and apple decay. In K-Means framework, Euclidean distance is utilized for discovery of the contaminated locale from the natural product picture and is changed over to RGB. The tone, figure and surface highlights are separated and include level combination are

completed for incorporating multiple highlights. At long last, Random woods classifier is executed for the grouping of plant infections.

Michael Gomez Selvaraj et al. [13] has proposed a paper on profound training is a new strategy for picture preparing then item identification with more prominent exactness in the grouping of different harvest illnesses. Move learning is one such well known methodology in profound realizing, where pre-prepared models are adjusted to do another undertaking. Prof. N. A. Auti et al. [14] applies profound learning method to distinguish and arrange the banana leaf infection. It likewise utilizes LeNet design as a convolutional neural organization to order picture informational collections. Seetharaman K. et al. [15] creators the combination of image processing and neural network techniques used for disease detection in banana fruit image. Jihen Amara et al. [16] distinguish two unmistakable banana illnesses in particular banana sigatoka and banana spot by applying profound neural organization innovation. The creator here has utilized LeNet engineering as a convolutional neural organization to group picture informational indexes. Basavaraj Tigadi et al. [17], creator has proposed programming answer for programmed plant sickness discovery lastly the rate contamination utilizing picture preparing method. Vipinadas. M. J et al. [18] proposed a framework which is an efficient module that distinguishes the Black sigatoka sickness and Panama shrivel infection on banana leaf. The infection reviewing has been finished utilizing ANFIS (Artificial Neural Network Fuzzy Interference System) classifier. At long last, classifiers examination has been performed utilizing disarray framework. A. Camargoa et al. [19] gives a different strategy to identify plant sicknesses utilizing picture handling procedure.

This framework is likewise an efficient module that recognizes the Black sigatoka illness and Panama shrivel infection on banana leaf. Karthik .G et al. [20] recognizes the contaminated leaves utilizing Economic Threshold Level (ETL) calculation. It identifies and forestalls the banana streak viral sickness utilizing Embedded Linux improvement board interfaced with a camera.

3. RESEARCH METHODOLOGY

The proposed framework presents a proficient strategy for picture design characterization in Banana stem disease detection. To start with, the stem illnesses on banana plant are identified utilizing normal versatile thresholding division, Linear PCA (L-PCA) and RNN classifier. At that point illnesses on the banana stem are evaluated utilizing RNN. At last, infections are named stem illness utilizing L-PCA and RNN. The proposed framework is appeared in figure 1.

Training

Banana stem image → Preprocessing using → Segmentation using average → Feature extraction using

Query → Preprocess → Segmenta → Feature

Extract

Test model

Feature extraction using → Classification

Classification → Classified
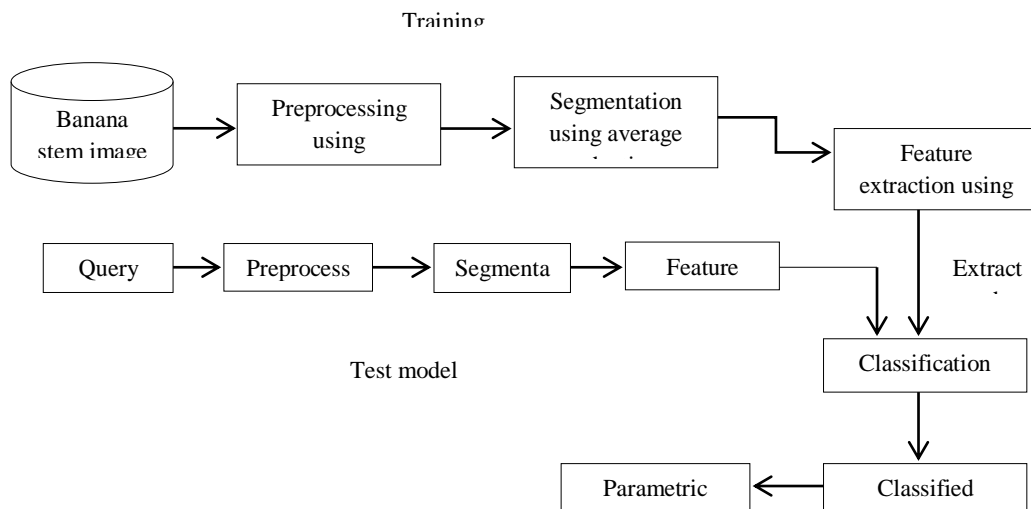
Classified → Parametric

Figure-1 Proposed Architecture

3.1 Dataset description

Our dataset involves around 18,000 field pictures of banana, gathered by banana specialists, from Diversity International (Africa) and Tamil Nadu Agricultural University (TNAU, Southern India). These field pictures were caught different natural conditions to construct a strong model. Our current CIAT banana picture archive comprises of roughly 18,000 genuine regional pictures.

The significant irritation (corm weevil) and illness class side effects as well as their preventive processes are introduced. Since side effects of various sicknesses and irritations are seen at various fragments of the banana vegetation, we caught pictures of every vegetation portions. Our ebb and flow archive was organized dependent on the sickness as well as the diseased vegetation fragments so each piece of the vegetation addresses a prototype.

3.2 Preprocessing using Morphological operation based on the pixel structuring

Morphology is an enormous scope of methods for picture acknowledgment which explore the effect based on shapes. An actualizing segment is added to information through numerical morphology, producing a yield picture as info. The estimation of every pixel thickness in a yield picture in a morphological activity depends on an examination of the comparing pixel in the information picture with its neighbors. Enlargement and disintegration are the most fundamental morphological tasks. Expansion unquestionably adds pixels to the limits of a picture's relics, while disintegration eliminates pixels from the edges of a picture. The picture quality embedded to or barred from a picture's ancient rarities relies upon size and design of the displaying factor being utilized addresses the picture. The state of some random pixel in the yield picture is determined during morphological expansion and disintegration activities by applying a standard to the comparing pixel and its neighbors in the information picture.

The standard utilized for preparing the pixels depicts expansion or disintegration as the technique.

MM1 dilation

For $p \in X$ do
For $e \in \text{SE}$ do
Image $(p + e) = 1$;
End for
End for

MM2 dilation

For $p \in \text{Fr}(X)$ do
For $e \in \text{SE}$ do
Image $(p + e) = 1$;
End for
End for

3.3 Segmentation using average adaptive thresholding

While navigating the picture, a rough moving normal of the last s pixels seen is registered. In the event that t is a percent below the normal estimation of the current pixel, it is set to dark, else it is set to white. This methodology works on the grounds that hard difference lines are kept up by contrasting a pixel with the normal of encompassing pixels and delicate inclination shifts are disregarded. The advantage of this interaction is that it needs just a solitary pass through the picture. For the estimation of s and 15 for the estimation of t, Wellner utilizes 1/eighth of the picture width. An issue with this methodology, nonetheless, is that it relies upon the pixels' checking request. Additionally, since the local examples are not consistently disseminated every which way, the moving normal is definitely not a reasonable portrayal of the encompassing pixels at each progression. We present an answer that doesn't experience the ill effects of these issues by utilizing the vital picture (and losing one extra cycle through the picture). Our strategy is spotless, clear, simple to code, and delivers a similar yield paying little mind to how the picture is handled. We ascertain the normal of the s x s window of pixels dependent on every pixel as opposed to figuring the last running normal of the s pixels appeared. For contrast, this is a superior normal since it thinks about adjoining pixels on the two sides. By utilizing the vital picture, the normal calculation is accomplished in straight time. During the initial pass through the information picture, we figure the essential picture. We measure the s x s normal in a subsequent pass utilizing the essential picture for every pixel in consistent time and afterward make the connection. On the off chance that t is not exactly this normal, at that point the estimation of the current pixel is set to dark, in any case the worth is set to white. The accompanying pseudo code shows our information picture strategy in, yield double picture out, width of picture w.

Procedure Average Adaptive Threshold (in,out,w,h)

for i = 0 to w do

sum ← 0

for j = 0 to h do

sum ← sum+in[i, j]

if i = 0 then

intImg[i, j] ← sum

else

intImg[i, j] ← intImg[i−1, j] +sum

end if

end for

end for

for i = 0 to w do

for j = 0 to h do

x1 ← i−s/2 {border checking is not shown}

x2 ← i+s/2

y1 ← j −s/2

y2 ← j +s/2

count ← (x2−x1)×(y2−y1)

sum ← intImg[x2,y2]−intImg[x2,y1−1]−intImg[x1−1,y2] +intImg[x1−1,y1−1]

if (in[i, j]×count) ≤ (sum×(100−t)/100) then

out[i, j] ← 0

else

out[i, j] ← 255

end if

end for

end for

3.4 Feature extraction using L-PCA

Linear Principal Component Analysis (L-PCA) is a mathematical technique that maps data from high-dimensional space to low-dimensional space using linear transformations. The low dimensional space of the covariance matrix can be calculated by Eigen vectors. The L-PCA

converts the properties of a dataset into new features called Linear Principal Components. A linear combination of the original variables of the dataset is the principal component. The primary components are categorized such that the first main component is the one that represents the highest variance in the information. In the dataset, the subsequent key components reflect the residual variance. These are not, however, correlated to the first main component of the data. Centred on the idea of eigenvalues and eigenvectors, Singular Value Decomposition (SVD) decomposes the original matrix into its components and this is used to eliminate the redundant features.

This transformation is defined in such a way that there is the greatest possible variance in the first linear principal variable. The PCA mathematical notion is as follows: for a given input central x_t where t = 1, 2,…, p, $x_t \in R^n$ and $\sum_{t=1}^{p} x_t = 0$, usually $n < p$, so that x is $n \times p$ input matrix. L-PCA converts this vector space into new vectors by solving the problem of Eigen values given in

$$\lambda_i \mu_i = c\mu_i \ i = 1,2,\dots,n$$

Where $\lambda_i \geq 0$ is one of the n Eigen values of the $n \times p$ covariance matrix c, where $c = (\frac{1}{p}(\sum_{t=1}^{p} x_t x_t^T))$, and μ_i is the Eigenvector whose corresponding to the Eigen value λ_i, then the PCs y_t are calculated as the orthogonal transformation of x_t;

$$y_t(i) = \mu_{1i}x_1 + \mu_{2i}x_2 + \cdots \dots \dots + \mu_{ni}x_n = \mu_i^T x_t \ \ i = 1,2,\dots,n$$

The key y t(i) part with the most elevated Eigen esteem addresses the vector in the informational collection with the most change. The PCs of the first vector space can be decreased to new vectors of their comparing Eigen esteems by utilizing just the initial not many Eigenvectors requested in plunging request of PCs k<n.

The calculation can be summed up in the accompanying five stages:

Stage 1: Compute the example covariance grid or, if accessible, utilize the genuine covariance network. The connection lattice instead of the covariance network is liked in certain circumstances.

Stage 2: Compute the Covariance/Correlation grid's straight principle segments and eigenvalues as characterized in the condition.

Stage 3: Pick the subspace measurement q and build the grid Aq out of A. This can be chosen by evaluating the amount of the information fluctuation is should have been saved. The proportion between the amount of the principal q eigenvalues and the amount of the multitude of eigenvalues is the held inconstancy.

Stage 4: Cluster the vectors q |V1 |,|V2 |,...,|Vn |∈ℜ to p ≥ q bunches. For the K-Means calculation, the distance count utilized is the Euclidean distance. In the event that a similar changeability as the PCA is required, it is for the most part important to choose p more noteworthy than q (normally 1-5 extra measurements are required)

Stage 5: Locate the comparing vector Vi for each group that is closest to the mean of the bunch. As an essential component, select the comparing highlight, xi. The decision of p qualities will yield this progression. There are two clarifications why the vector nearest to the mean is picked. This component can be viewed as the centre element of that group, the most

predominant aspect in that bunch, which holds the most un-excess element subtleties in different groups. Consequently, both of the properties we needed to accomplish are fulfilled enormous "spread" in the lower dimensional space, and the first information is all around addressed.

3.5 Classification using recurrent neural network (RNN)

For processing the data sequence $x(t) = x(1),\dots,x(\tau)$ in specific Recurrent Neural Network (RNN) is used step-t limits from 1 to τ by their time index. The sequential input tasks consist of data obtained from the clinic where it was gathered in the database. We use RNN for this set of features here. This selection process utilizes RNN to train and test the data sequence, since the attributes have already been predicted. RNN is known as repeating because all the data sequence for each variable where the output was centred on the earlier training and testing phase was trained for the similar task.
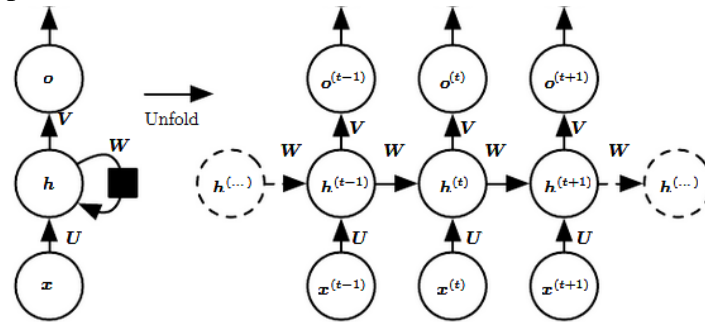

Figure 2-Architecture of RNN

3.5.1 Architecture of RNN

The figure above shows RNN's architecture. The symbolization of the left side gives the image of the RNN and the right side as it is unfolded, which would be the complete RNN network. Unfolding represents a complete sequence of the network. When the sequence has a data sequence of 3 attributes, then it is delivered into an unfolded 3-layer neural network, each layer has each input data attribute.

Input: the network input was taken as x(t)where their time phase is t. If x1 has a single vector, it will have unique data for its corresponding mission.

Hidden state*: Along with the network memory act, the hidden state was represented by h(t) with time t. The hidden data h(t) has been determined as hidden data h(t) based on current input along with the earlier stage of the hidden state: h(t) = f(U x(t) + W h(t−1)). Non-linear function transformation f is known to be tanh, ReLU.

Weights: The characterized correlations hidden with the RNN input were given as the weight matrix U, the characterized correlations with the recurrent hidden-to-hidden input given by the weight matrix W, the characterized correlations with the hidden-to-output output given by the weight matrix V, all the weights together are known to be shared over time.

Output: The output of the network is given as o(t). The above architectural figure has also been included in the downstream, the outward arrow after o(t) often exposed to non-linearity in particular additional layers.

Forward Pass

The activation operation for the hidden units is not specified in the architecture above. When this process is about to continue, few assumptions have been made here: 1) for hidden layers, we assume that the activation operation is hyperbolic tangent. 2) Because RNN was used in the prediction class, it was necessary to assume that the output was discrete. The usual method of representing the discrete variables is necessary to represent the prediction class of the data sequence by RNN, taking into account the output o (t) by representing the log probability of un-equalization in discrete parameters for each probable value. To obtain the vector of equalizing probabilities where the softmax function has been applied to the output.

The forward pass for RNN is defined by the following equation.

$$
\begin{aligned}
\boldsymbol{a}^{(t)} &= \boldsymbol{b} + \boldsymbol{W}\boldsymbol{h}^{(t-1)} + \boldsymbol{U}\boldsymbol{x}^{(t)} \\
\boldsymbol{h}^{(t)} &= \tanh(\boldsymbol{a}^{(t)}) \\
\boldsymbol{o}^{(t)} &= \boldsymbol{c} + \boldsymbol{V}\boldsymbol{h}^{(t)} \\
\hat{\boldsymbol{y}}^{(t)} &= \mathrm{softmax}(\boldsymbol{o}^{(t)})
\end{aligned}
$$

The input sequence was mapped to the output sequence, which is given as an example of RNN where the sequence length is identical. The total loss of x values was combined with the sequence of y values for the given sequence, where their total loss was above the entire time phase. Then, via the argument for softmax action, the output o(t) was presumed to be used to acquire the vector probabilities for the output as ŷ. The loss has been assumed for the input given when the log-likelihood L is negative for the original target y(t). Below the architecture is given.
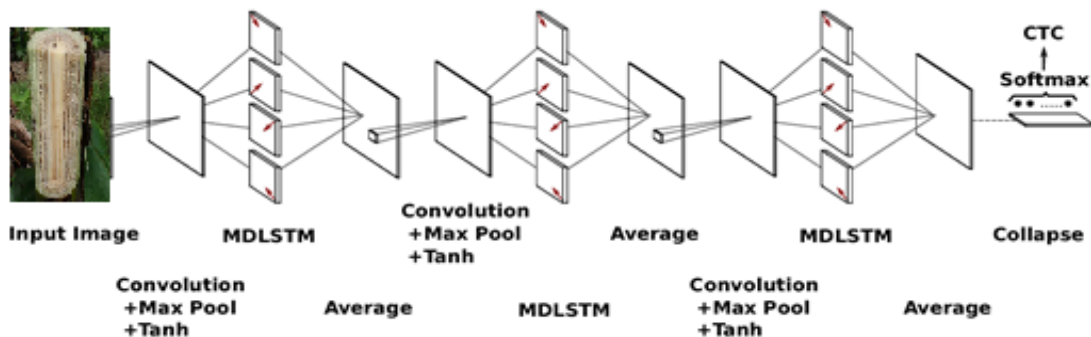

Figure-3 Classification architecture for RNN

Computing Gradients:

The provided loss operation is L, the gradients needed to be evaluated by the weight matrices defined as U, V, W, and bias terms b, c, the rate of learning alpha has also been told. The normal back propagation is the same as the gradient, which gives the sense of loss by changing weight characteristics. The W weights that have been adjusted to decrease the loss by the equation provided below,

$$W \leftarrow W - \alpha \frac{\partial L}{\partial W}$$

The other weights U, V, b, and c have also been preceded by a similar technique. For the RNN equations above, the gradient computation via BPTT was completed. The metrics U, V, b, and c were indexed by t for x(t),h(t),o(t) and L by the graph of the computation of the nodes along with the node sequence (t). On the basis of the node computation followed by the graph, we need the gradient computation ∇nL repeatedly for each node. Gradient has been used as the softmax operation statement in conjunction with the output o (t) and can acquire the vector likelihood ŷ for the output here. For the original assignment y, the assumption of loss function was taken as negative log likelihood (t).

$$(\nabla_{o^{(t)}} L)_i = \frac{\partial L}{\partial o_i^{(t)}} = \frac{\partial L}{\partial L^{(t)}} \frac{\partial L^{(t)}}{\partial o_i^{(t)}} = \hat{y}_i^{(t)} - \mathbf{1}_{i=y^{(t)}}$$

From the diagram below, we can clearly see that state of hidden h (t) has flow in gradient from present yield as well as following state of hidden at time t together.
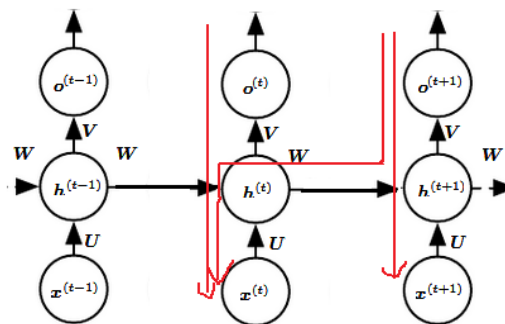

Figure- 4 Gradients of RNN

The time step τ at end, h (τ) only has o(τ) as a descending, consequently it has simple gradient:

$$\nabla_{\boldsymbol{h}(\tau)} L = \boldsymbol{V}^\top \nabla_{\boldsymbol{o}(\tau)} L$$

In time, we can then repeat backwards to gradients of backpropagation by interval, from t=τ −1 down to t = 1, notice that h (t) (for t < τ) has o (t) and h(t+1) as descendants together. In this way, their gradient is expressed as follows:

$$\nabla_{\boldsymbol{h}^{(t)}} L = \left(\frac{\partial \boldsymbol{h}^{(t+1)}}{\partial \boldsymbol{h}^{(t)}}\right)^\top (\nabla_{\boldsymbol{h}^{(t+1)}} L) + \left(\frac{\partial \boldsymbol{o}^{(t)}}{\partial \boldsymbol{h}^{(t)}}\right)^\top (\nabla_{\boldsymbol{o}^{(t)}} L)$$

$$= \boldsymbol{W}^\top \mathrm{diag}\left(1 - \left(\boldsymbol{h}^{(t+1)}\right)^2\right)(\nabla_{\boldsymbol{h}^{(t+1)}} L) + \boldsymbol{V}^\top (\nabla_{\boldsymbol{o}^{(t)}} L)$$

Once the gradients on the computational graph's internal nodes are collected, we can obtain the gradients on the nodes of the parameters. For all parameters, the gradient calculations which use the chain rule are:

$$\nabla_{\boldsymbol{c}} L = \sum_t \left(\frac{\partial \boldsymbol{o}^{(t)}}{\partial \boldsymbol{c}}\right)^\top \nabla_{\boldsymbol{o}^{(t)}} L = \sum_t \nabla_{\boldsymbol{o}^{(t)}} L$$

$$\nabla_{\boldsymbol{b}} L = \sum_t \left(\frac{\partial \boldsymbol{h}^{(t)}}{\partial \boldsymbol{b}^{(t)}}\right)^\top \nabla_{\boldsymbol{h}^{(t)}} L = \sum_t \mathrm{diag}\left(1 - \left(\boldsymbol{h}^{(t)}\right)^2\right) \nabla_{\boldsymbol{h}^{(t)}} L$$

$$\nabla_{\boldsymbol{V}} L = \sum_t \sum_i \left(\frac{\partial L}{\partial o_i^{(t)}}\right) \nabla_{\boldsymbol{V}^{(t)}} o_i^{(t)} = \sum_t (\nabla_{\boldsymbol{o}^{(t)}} L) \boldsymbol{h}^{(t)\top}$$

$$\nabla_{\boldsymbol{W}} L = \sum_t \sum_i \left(\frac{\partial L}{\partial h_i^{(t)}}\right) \nabla_{\boldsymbol{W}^{(t)}} h_i^{(t)} = \sum_t \mathrm{diag}\left(1 - \left(\boldsymbol{h}^{(t)}\right)^2\right)(\nabla_{\boldsymbol{h}^{(t)}} L) \boldsymbol{h}^{(t-1)\top}$$

$$\nabla_{\boldsymbol{U}} L = \sum_t \sum_i \left(\frac{\partial L}{\partial h_i^{(t)}}\right) \nabla_{\boldsymbol{U}^{(t)}} h_i^{(t)} = \sum_t \mathrm{diag}\left(1 - \left(\boldsymbol{h}^{(t)}\right)^2\right)(\nabla_{\boldsymbol{h}^{(t)}} L) \boldsymbol{x}^{(t)\top}$$

4. RESULTS AND DISCUSSION

The simulation outcomes for banana stem disease detection from the input dataset are discussed in this section. Various outputs have been discussed below: the dataset folder was originally created where the various banana fruit images are available for the testing process.

The fruit disease indication was established with greater precision and was also adequately trained and collected in the dataset. The images of banana plants affected by disease have been selected and provided for identification as an input. Owing to the irregular dimensions of the pre-set dimension, this image was then resized to precise dimensions. This resized image was translated from an RGB image to a grayscale image of the stem. This image is therefore pre-processed using pixel structuring-based morphological activity and segmented using average stem adaptive thresholding. Linear Principal Component Analysis

(L-PCA) for the stem is used to extract the element. Using recurrent neural network, then graded (RNN). The steps are collecting stem sample is given in below figure 5.

Figure 5- Sample Collection.

Figure 5- Procedure for sample collection from suspected infected banana plants by Fusarium oxysporum cubense in disease free areas. A y B Cut of a pseudo stem fragment. C. View if the pseudo stem fragment severed showing necrotic vascular strands. D. Fragments of affected tissue inside of a flask or envelop hermetically closed. E. Dissected vascular strands of pseudo stem showing necrosis caused by pathogen. F and G. Sampled plant with reposition of the fragment cut in the original place and covered by adhesive tape to avoid exposition of tissues and exudates at environment.

Table-1 Parametric analysis for proposed technique

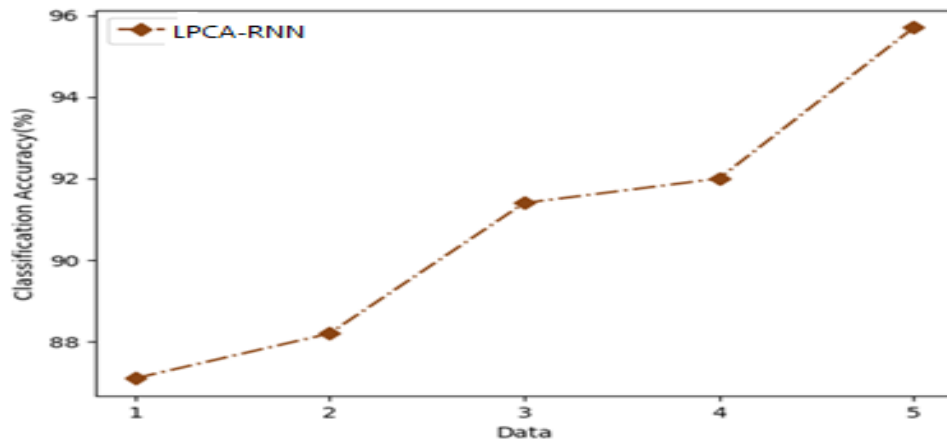| Input image	Accuracy	Precision	Recall	F1 - Score
1	87.1	85.4	84.7	87.3
2	88.2	86.6	90.3	88.3
3	91.4	88.3	91.6	90.7
4	92	89.6	94.4	92.2
5	95.7	90	94.6	91.3

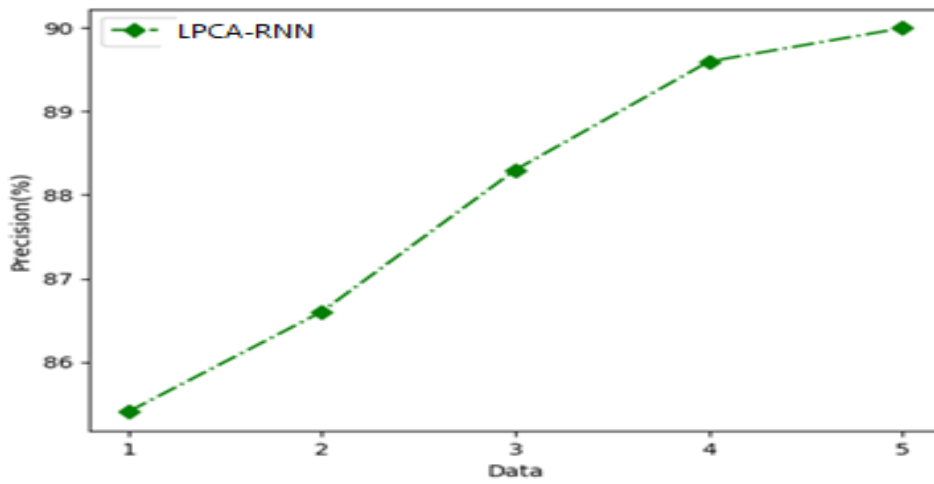Figure 6- Classification accuracy for LPCA-RNN
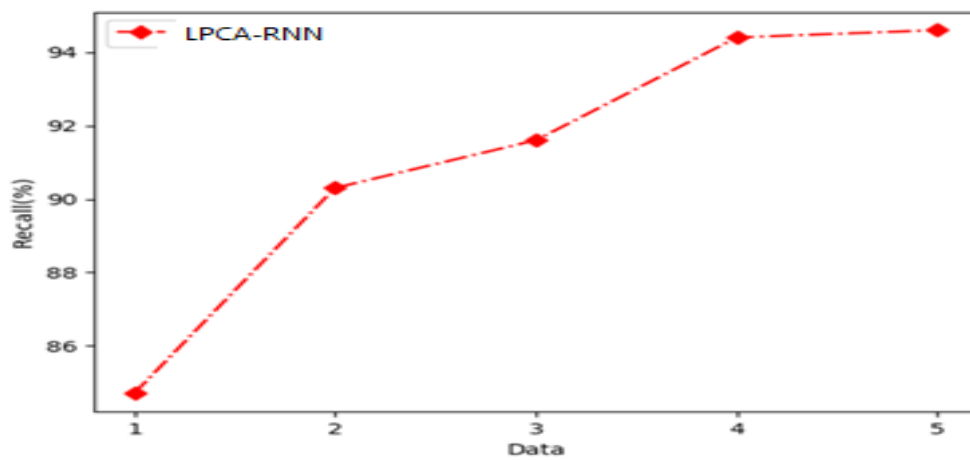

Figure 7- Precision for LPCA-RNN
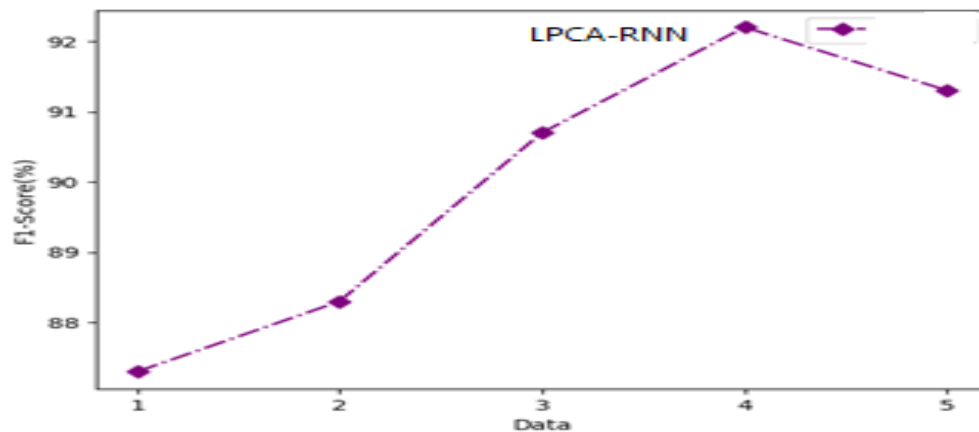

Figure 8- Recall for LPCA-RNN

Figure-9 F1 score for LPCA-RNN

The above table 1 shows the parametric analysis for proposed LPCA-RNN. The figure 6,7,8,9 shows the graphical representation of proposed LPCA-RNN for table 1 and the figure 10 shows disease detected part of banana stem.

Figure-10 Disease detection in stem of banana tree

5. CONCLUSION

Numerous PC visioned approaches have been reported for computerized crop infection location and arrangement; however there is as yet a slack in detail investigation of continuous recognizable proof of bugs and illnesses. An epic strategy for utilizing the straight PCA-RNN technique was investigated in this paper to identify banana bug and infection indications naturally on various pieces of banana plants utilizing continuous field manifestations of the picture. This framework gives a sensible and appropriate methodology for the recognizable proof of the class and position of illnesses in banana plants, which is the key differentiation that is practically identical to different techniques for arrangement of plant infections. For different banana illnesses, the set up model had the option to distinguish the distinction among solid and tainted plant areas. The thorough models worked from this examination would be more valuable for the improvement of a choice emotionally supportive network to help with the early identification and the executives of pests and illnesses.

References

1. Selvaraj, Michael Gomez, et al. "Detection of banana plants and their major diseases through aerial images and machine learning methods: A case study in DR Congo and Republic of Benin." *ISPRS Journal of Photogrammetry and Remote Sensing* 169 (2020): 110-124.

2. Amara, Jihen, Bassem Bouaziz, and Alsayed Algergawy. "A deep learning-based approach for banana leaf diseases classification." *Datenbanksysteme für Business, Technologie und Web (BTW 2017)-Workshopband* (2017).

3. Owomugisha, Godliver, et al. "Automated vision-based diagnosis of banana bacterial wilt disease and black sigatoka disease." *International Conference on the Use of Mobile ICT in Africa*. 2014.

4. Ferentinos, Konstantinos P. "Deep learning models for plant disease detection and diagnosis." *Computers and Electronics in Agriculture* 145 (2018): 311-318.

5. Al-Daour, Ahmed F., Mohammed O. Al-Shawwa, and Samy S. Abu-Naser. "Banana Classification Using Deep Learning." (2020).

6. Sandip. P Bhamareatel., "Detection of Black Sigatoka on Banana Tree using Image Processing Techniques". IOSR Journal of Electronics and Communication Engineering (IOSRJECE) ISSN: 2278-2834, ISBN: 2278-8735, PP: 60-65 www.iosrjournals.org.

7. Tigadi, B., & Sharma, B. (2016). "Banana Plant Disease Detection and Grading Using Image Processing". International Journal of Engineering Science, 6512.

8. Amara, J., Bouaziz, B., & Algergawy, A. (2017). "A Deep Learning-based Approach for Banana Leaf Diseases Classification". In BTW (Workshops) (pp. 79- 88).

9. Zhang, S. W., Shang, Y. J., & Wang, L. (2015). "Plant disease recognition based on plant leaf image". The journal of animal & plant sciences, 25(3 Supplement 1), 42-45.

10. Owomugisha, G., Quinn, J. A., Mwebaze, E., & Lwasa, J. (2014). "Automated Vision-Based Diagnosis of Banana Bacterial Wilt Disease and Black Sigatoka Disease". In International Conference on the Use of Mobile ICT in Africa.

11. P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," Mach. Learn., vol. 63, no. 1, pp. 3– 42, 2006.

12. Samajpati, B. J., & Degadwala, S. D. (2016, April). Hybrid approach for apple fruit diseases detection and classification using random forest classifier. In 2016 International Conference on Communication and Signal Processing (ICCSP) (pp. 1015-1019). IEEE.

13. Michael Gomez Selvaraj, Alejandro Vergara, Henry Ruiz, Nancy Safari, Sivlingam Elavabalan Walter Ocimati& Guy Blomme, "AI-powered banana disease and pest detection".

14. Prof. N. A. Auti, Shraddha Kadam, Sneha Satav, Poonam Bhapkar, Shraddha Dhumal, International Journal of Innovative Research in Engineering & Multidisciplinary Physical Sciences (IJIRMPS), Volume 6, Issue 6, 2018.

15. T. Mahendran, K. Seetharaman, (2023). "Feature Extraction and Classification Based on Pixel in Banana Fruit for Disease Detection Using Neural Networks", IEEE *Xplore* (ICAECT) ISBN:978-1-6654-9401-4, DOI: 10.1109/ICAECT57570.2023.10117959.

16. Jihen Amara, Bassem Bouaziz, Alsayed Algergawy, "A Deep Learning-based Approach for Banana Leaf Diseases Classification", B. Mitschang et al. (Hrsg.): BTW 2017 – Workshop band, Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2017.

17. Basavaraj Tigadi, Bhavana Sharma, "Banana Plant Disease Detection and Grading Using Image", IEEE Autotestcon Year: 2016, Research Article Volume 6 Issue number 6.

18. Vipinadas. M. J, A. Thamizharasi, "Detection and Grading of diseases in Banana leave using Machine Learning", International Journal of Scientific & Engineering Research, Volume 7, Issue7, July-2016.

19. Camargoa, J.S. Smith b, "An image-processing based algorithm to automatically identify plant disease visual symptoms".

20. Karthik .G, Praburam. N, "Detection And Prevention Of Banana Leaf Diseases From Banana Plant Using Embedded Linux Board", Publisher: IEEE.