BioGecko

Vol 12 Issue 02 2023
ISSN NO: 2230-5807

MCIP: Mining Crop Image Data On Pyspark Dataframe Using Pca And K-Means

Yashi Chaudhary[1], Heman Pathak[2]
[1] Research Scholar,Gurukul Kangri (Deemed to be University), Haridwar, Uttarakhand, India
[1]*mohita.chaudhary5@gmail.com*
[2] Professor,Gurukul Kangri(Deemed to be University), Haridwar, Uttarakhand, India

Abstract

In India, crop related diseases and pests result in annual losses of more than $500 billion. The primary contributor to the $500 billion in losses is leaf blight. Farmers that raise forage and grain sorghum suffer the most. This disease affects crops like Maze, Rice, Tomato, Potato, Millet, and Onion. Early identification and severity assessment of the disease in plants can help reduce this loss. But it is challenging due to variances in crop species, crop disease symptoms, and environmental variables. The existing techniques are not generalized to classify and predict the disease. All the techniques work on a known data with a known output. The existing techniques pre-process the images and segment them to extract the relevant features. Segmentation requires pre-processing, dilation, and edge detection. This results in loss of the vital information leading to incorrect classification. Moreover, the techniques used thus far do not test algorithm on niche data. Deep learning techniques are vulnerable to overfitting. We propose Mining Crop Image data using PySpark (MCIP) data frame. MCIP uses Principal Component Analysis (PCA) to pick relevant features. The collected PCA features are then used to identify homogeneous subgroups using K-means. The categorized predictive output aids in detecting the diseases in the leaves of Potato. MCIP is not limited to potato only, it can be used to detect disease in the leaves of any crop. To ascertain our claim, we tested MCIP on rice disease dataset. We evaluated MCIP's Accuracy, Silhouette score, speed and F1 score to check its robustness. MCIP outperformed prior approaches in terms of speed and accuracy. The accuracy is amazingly close to 100 percent.

Index Terms— Big Data, Data Mining, K-means, PCA, PySpark, Agriculture

I. INTRODUCTION

A country's economy is primarily reliant on its agriculture. Food self-sufficiency is essential for a country's survival. While researchers and growers concentrate on a range of factors to increase productivity, crop loss due to disease is one of the most critical problems they face. Crop growth monitoring and early pest infestation identification remain a challenge. Manual intervention to monitor and identify insect and pest infestations is getting increasingly challenging as farming expands to larger fields (Velusamy et al. 2021;Dhaliwal GS et al.2021).Using satellite images to identify crop loss at the field parcel scale is difficult for several reasons: first, crop loss is caused by a variety of factors throughout the growing season; second, reliable reference data on crop loss is lacking; and third, crop loss can be defined in a variety of ways (Hiremath S et al.2021). Helminthophobia The fungus turcicum Pass is responsible for leaf blight. On sorghum leaves, the disease emerges as reddish-purple or brown spots that aggregate into large lesions, especially in moist climates. It is equally destructive to seedlings as mature plants. The disease is frequently detected in a moderate to severe form on forage sorghum in the Indian states of Haryana, Rajasthan, Uttarakhand, and Uttar Pradesh(Muimba-Kankolongo A. 2018).Because of considerable damage to photosynthetic machinery on the leaf, the disease can occasionally become pandemic, affecting forage productivity, quality, and grain yield. During severe epidemics, grain yields might drop by as much as 50% or more (Das IK et al. 2016).

Plant health monitoring and early detection of symptoms are required to limit disease transmission, which aids farmers in effective management methods and increases output. As a result, crop disease identification is critical to maintaining agricultural output. Traditional plant disease diagnosis procedures rely on the farmer's experience, which is inherently inaccurate and imprecise. Earlier, researchers used a spectrometer to determine if plant leaves were healthy or sick (Sasaki Y et al. 1998). Another way was to use the polymerase chain reaction (Henson JM et al. 1993) or real-time polymerase chain reaction (Koo C at al. 2013) to extract the (Deoxyribonucleic acid) DNA from the leaves. The authors of (Prasad S et al. 2016) described a method for identifying plant leaf diseases using soft computing. To identify and categorise plant leaf diseases, the authors utilised a genetic algorithm for image segmentation. The suggested approach was evaluated on a variety of plant leaves and shown to be effective in identifying illnesses early on. The authors of (Ali H et al. 2017) investigated the identification of plant diseases using a pattern recognition algorithm to calculate crop pictures. To identify plant diseases the Gabor Wavelet

Transform (GWT) approach was used with pattern recognition. In (Prasad S et al. 2016) an innovative concept for disease detection termed automated mobile vision was introduced. To identify diseases in plants, the authors utilised a hybrid approach termed GWT- (Gray-Level Co-Occurrence Matrix) GLCM. Citrus fruit diseases might be precisely recognised using the DeltaE approach, according to the authors of (Ali H et al. 2017). The researchers used KNN and cubic SVM to classify diseases based on image level and disease level in their study.

Several approaches for detecting leaf diseases in various crops have been developed in recently (Singh UP et al. 2019;Zhang K et al. 2019;Lu J et al. 2017). In majority of the strategies, image processing techniques were used to extract features, which were then input into a classification technique. (Deepa NR et al. 2021) suggested a method for detecting plant leaf disease. The authors used the Kuan filter to remove noise before extracting colour, shape, and texture information using the Hough transformation. The plant leaf disease was classified using a reweighted linear programme boost classification. The suggested technique's performance was assessed using the PlantVillage dataset.(Hamuda E et al. 2018) suggested a crop identification system based on image processing that used the Kalman filtering algorithm and the Hungarian algorithm. (Mahumd M et al. 2018) assessed significant applications of hierarchical learning, reinforcement learning, and deep reinforcement learning techniques, comparing their performance based on network design, feature selection, and learning, as well as parameter optimization. Over a mobile acquired picture,(Picon A et al. 2019) employed the ResNet-50 architecture, which is a deep CNN architecture. They used stochastic gradient descent optimization to train the network.(Ferentinos KP 2018) suggested a deep learning-based approach for detecting plant leaf disease using multiple CNN architecture models on an open dataset with 58 discrete classifications. (Huang T et al. 2018) proposed utilizing the RBF (radial basis function) kernel of a support vector machine to identify sugarcane borer illness. The choice is made in a quick manner utilizing basic processors in this technique. It also uses less memory for data storage, i.e., data collected during the training process.

There are various datasets on which classification and prediction of disease is done. Advances in artificial intelligence have aided researchers in identifying and diagnosing plant disease utilising proper image processing and machine learning methodologies.(Singh UP et al. 2019) classified using mango leaves using CNN. (Singh V.2019) used image segmentation and Particle Swam Optimization (PSO). Convolutional Neural Network (CNN) and Deep CNN (DCNN) is widely used to classify and predict leaf diseases of wheat, tomato, corn, and seasonal crops (Sladojevic et al. 2016; Sharma P et al.2020; Agarwal M et al. 2020; Mishra S et al. 2020; Khamparia A et al. 2020; Hussain A et al. 2018). Researchers have also used Deep Neural Networks (DNN) to classify and detect the plant leaf diseases (Venkataramanan A et al. 2019). Deep learning has also attracted researchers of (Chandy A. 2019; Karthik R et al.2020; Zhang Y et al. 2020). (Tarik MI et al. 2021) used Image Processing picture division with machine learning for potato leaf disease detection. More finding of the related work is summarized in table 1.

Table 1:Summary of related work

Paper	Dataset	Technique	Advantage	Disadvantage
(Khamparia A et al. 2020)	Potato	Deep CNN	Detects multiple diseases. Achieves an accuracy of 96.46%	Requires large dataset and GPU. It is very expensive to train.
(Nazki H et al. 2020)	Tomato leaves	Activation Reconstruction loss Generative Adversarial Network (ARL-GAN) and CNN.	Demonstrates synthetic image information clearly. Improves classification. Achieves an accuracy of 87.6%.	The procedure is complex and costly to get desired results. There is an issue of mode collapse too.
(Ganatra N et al. 2020)	Plant Village	ResNet 50 and 101	Return very high accuracy of disease classification with lesser layers. The accuracy achieved is 99.7%	The model works for certain epochs but on increasing the epochs it would suffer from overfitting problem thus resulting in reduced accuracy.
(Sambasivam G et al. 2021)	Cassava	CNN	Detects multiple diseases. Achieves an accuracy of 93%	Suffers from overfitting problem. Large training dataset

				is needed. Position and orientation of an object is not encoded.
(Geetharamani G et al. 2019)	Maize, Potato, Tomato	CNN and Autoencoders	Can detect multiple diseases. Achieves an accuracy of 97.50%	Suffers from overfitting problem on a larger dataset. Requires big clean data to arrive at desired results.
(Liang q et el. 2019)	Potato	Resnet 50	Achieves an accuracy of 98%	Better techniques with 99.7% accuracy are available.
(Khalifa NE et al. 2021)	Potato	CNN	Detects early and late phases of blight. Achieves a 98 % overall accuracy	Limited to small and specific dataset.
(Rozaqi AJ et el. 2020)	Potato	CNN	Early and late blight are identified using an accuracy of 92%.	[38] shows better results.
(Sanjeev K et al. 2020)	Potato	Feed Forward Neural Network (FFNN)	Early Blight and Late blight are detected with an accuracy of 96.5%.	Loses neighbourhood information as it cannot move back and learn.
(Barman U et al. 2020)	Potato	Simplified Bayesian CNN (SBCNN)	Early Blight and Late blight are detected with an accuracy of 96.75%.	The model requires more parameters to train.
(Jhonson J et al. 2021)	Potato	Mask R-CNN	Early Blight and Late blight are detected with an accuracy of 98%.	Limited to a single dataset only.
(Lee TY et al. 2020)	Potato	CNN	Early Blight and Late Blight can be found with 99% accuracy.	The model requires lot training data. It also does not encode the position and orientation of the leaf.
(Islam M et al. 2017)	Potato	Segmentation, and Multi SVM	Early Blight and Late blight are detected with an accuracy of 95%.	It is not suitable for large dataset. Segmentation could result in loss of features.
(Rashid J et al. 2021)	Potato	Yolov5 Segmentation, Deep learning using CNN	Early Blight and Late blight are detected with an accuracy of 99.75%.	The technique is limited to Early blight and late blight detection of a single dataset. Overfitting problem is not addressed properly which is very common issue with deep learning.

According to our survey the techniques are difficult, costly, and time-consuming, and they need a highly professional operation, extensive experimentation, and extensive use of crop protection agents. The existing models have trained, tested, and validated on benchmark datasets. The datasets have limited images. None of the technique addresses to the big problem of time taken in training and testing large image dataset. The detections are limited to few diseases. If a foreign leaf data without a label is introduced the techniques would not return the desired results. MCIP tries to address these issues along with the issue of speed by using Spark framework.

Spark is one of the most popular new technology trends. It's the framework with the best chance of realising the benefits of the combination of Big Data and Machine Learning (Jonathan J et al. 2021). It's fast (up to 100x quicker than typical Hadoop MapReduce thanks to in-memory operations), delivers robust, distributed, fault-tolerant data objects (referred to as RDDs), and combines perfectly with the realms of machine learning and graph analytics. Pyspark is an application programme interface (API) that connects spark with python.

Multivariate data tables may be found on leaves. In varying quantities, the leaves might be exceedingly healthy or disease-ridden. The severity ranges from low to severe. PCA is a statistical process that summarizes content of large data tables by having a lower number of "summary indices" that can be displayed and analyzed more readily. It aids in the identification of trends, leaps, clusters, and outliers. PCA is used by MCIP to extract features from a dataset. To categorize the clusters, trends, and outlier features generated by PCA, MCIP employs the K-means clustering method.

MCIP is independent of dataset and can take any image dataset and classify it. We have calculated Silhouette score to understand how accurately images are classified. Clustering algorithms such as K-Means uses the silhouette score to look at how well samples are grouped together with other samples that are alike. The Silhouette score is calculated for each sample of unique clusters (Dutta P et al. 2021).

II. Hardware setup

Since the proposed work requires parallel processing, the system requires Nvidia graphic card and decent memory. We tested the proposed model using gaming laptop from HP with 8 Gb RAM, Intel core i5, 10th generation processor.

A. Dataset

Benchmark datasets of potato leaves and rice leaf diseases [5] dataset is taken. Potato leave dataset contains 4962 images distributed in training, testing, and validation folders. The image folders are marked as late blight, early blight and healthy. Rice leaf data set contains 5932 images marked as bacterial blight, blast, tungro and brown spot. The dataset does not have training, testing, and validation dataset. The images are distributed in the respective disease folders.

Table 2: Potato leaf diseases

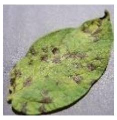
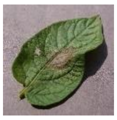
 Early Blight	 Late Blight	 Healthy

Table 3: Potato Leaf Dataset

Class	No of images
Early Blight	1928
Late Blight	1714
Healthy	1320
Total	4962

Table 4: Rice leaf diseases

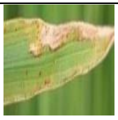
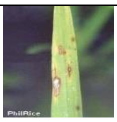
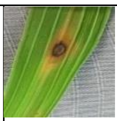
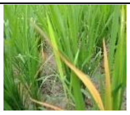
 Bacterial Blight	 Blight	 Brown Spot	 Tungro

Table 5: Rice Leaf Dataset

Class	No of images

Bacterial Blight	1584
Blast	1440
Brown Spot	1600
Tungro	1308
Total	5932

B.Objective function

The objective of the research is to accurately identify the disease. The function is mathematically represented here:

$$D = \left\{ P \left\{ \sum_{i=1}^{n} L_i \xrightarrow{create\ dataset} \{L_d\} \rightarrow PCA\{L_d\} \rightarrow K-means\{L_{PCA}\} \rightarrow classify\left(L_K\right) \right\} \right\} \ldots.\ eq(1)$$

Here,

P is predictive analysis
L_i is leaf
L_d is leaf data
L_{PCA} is PCA features
L_K is K-means
D is the predicted disease

The objective function is summarized as:

1. Read each folder name and data.
2. Normalize each image data by dividing them by 255.
3. Store the normalized data with labels in a list and convert it into a .csv file
4. Read the .csv file and drop all the Nan values.
5. Apply PCA on the data and get PCA features.
6. Create a K-means model.
7. Standardize PCA features using K-means model
8. Compile (fit) PCA features and standardized features with K-means model.
9. Transform output from 8 for predictive analysis
10. Take a new leaf
11. Extract features using PCA
12. Using transformed output features of the new leaf are passed to get the disease.

C.Mathematical formulation for computing the results

MCIP uses clustering techniques for classification and prediction. To check the goodness, we calculated the Silhouette score. The value of Silhouette ranges between -1 and 1.

$$Silhouette = \frac{A_{ic} - A_{inc}}{max\left(A_{inc}, A_c\right)} \ \ldots\ldots.\ eq(2)$$

Here,

A_{ic} is Average inter – cluster

A_{inc} is Average intra – cluster

For checking the goodness of classification of MICP we calculated Precision, Recall, Accuracy, and harmonic mean or F1 score. The calculations are based on percentage of correctly identifies positives or True Positive (TP), correctly identifies negatives or True Negative (TN), identified as positive but not positive or False Positive (FP), and identified as negative but not negative or False Negative (FN).

1) Precision

$$Precision = \frac{\# TP}{(\# TP + \# FP)} \text{... eq(3)}$$

2) Recall

$$\text{Recall} = \frac{\# TP}{(\# TP + \# FN)} \text{... eq(4)}$$

3) Accuracy

$$Accuracy = \frac{(\# TP + \# TN)}{(\# TP + \# TN + \# FP + \# FN)} \text{... eq(5)}$$

4) F1 Score

$$F1 = 2 \times \frac{(\text{Precision} \times \text{Recall})}{(\text{Precision} + \text{Recall})} \text{... eq(6)}$$

III. Research Methodology

The model is graphically represented to give an overview of the working. Detailed implementation is explained through an algorithm.

A. Model

Model is graphically represented in Fig1(A) and 1(B).

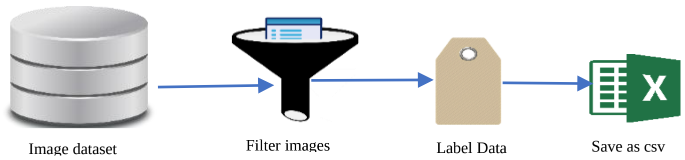


Fig.1(a) Create Dataset

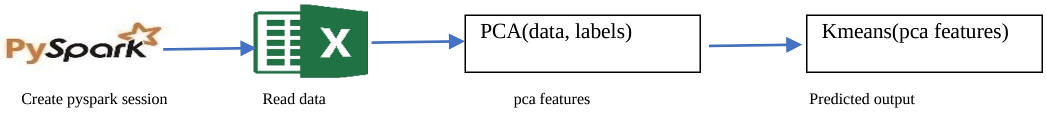


Fig .1(b) Apply PCA and K-Means on data in PySpark environment

The proposed model is represented using Fig. 1(a) and Fig.1(b). MCIP reads the data from each folder and then filters it to scale the image evenly. The images are then labeled with their sub-folder names. The digitized images are then stored as a .cave file. To extract the features using PCA, we setup a PySpark environment. The data is read in Pyspark framework. Once the data is read PCA is applied on the data with labels to extract the PCA features. Finally, K-means is applied on PCA features to classify and predict the data.

B. Implementation

Before implementing PCA and K-means to classify and detect the disease we calculated the value of k using elbow method. The elbow technique clusters the dataset using k-means for a range of k values (for example, 1-32), and then computes the average score for all clusters for each value of k. By default, the distortion score is computed, which is the sum of the square distances between each point and its assigned centre. MCIP reshapes the images into 32X32 images. For estimating the value of k we passed 1024*n data to the elbow visualizer. Where n is the number of images and 1024 is the number of pixels in an image. Potato dataset has three classes which are numbered 0, 1, and 2.

Before implementing PCA and K-means to classify and detect the disease we calculated the value of k using elbow method. The elbow technique clusters the dataset using k-means for a range of k values (for example, 1-32), and then computes the average score for all clusters for each value of k. By default, the distortion score is computed,

which is the sum of the square distances between each point and its assigned centre. MCIP reshapes the images into 32X32 images. For estimating the value of k we passed 1024*n data to the elbow visualizer. Where n is the number of images and 1024 is the number of pixels in an image. Potato dataset has three classes which are numbered 0, 1, and 2.

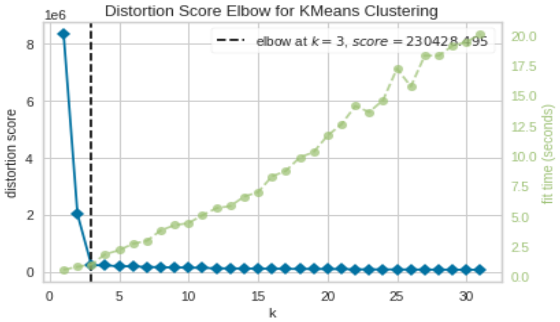


Fig. 2. Plot of Elbow method

In fig.2 ,x-axis has different values of k and y axis has distortion score s1^{e6} here, s is the score. The score is of all the pixels of the image dataset. Elbow method The Elbow technique identifies k clusters that best reflect the data points in their individual clusters. As a result, the distance between the data points and their respective cluster centroids would be the assessment measure. In the plot the bending starts at 3. This is the point where the reduction in Root Mean Square Error (RMSE) is no longer big enough to justify raising the k value.
The equation can be represented mathematically:

$$k = min\left(RMSE\left(\sum_{i=1}^{n-1}\sum_{j=1}^{n}\frac{\left\|d_i - d_j\right\|}{2}\right)\right) \quad \ldots\ldots eq\cdot(7)$$

Here,
n is the number of clusters
d_i and d_j are the distances between two data points

The value of k we get here is 3 which is also the number of classes in potato leaf dataset. This is used in PCA for number of components and k-means clustering.

1) *Algorithm : Classification*

```
Setup:
        Read dataset
        Install required modules
        Create spark session
        Create context

   //SparkContext is the entry gate of Apache Spark functionality.
Start:
Step 1. Prepare the dataset
     validation_datagen ← Using Image Data Generator rescale validation image
     test_datagen ← Using Image Data Generator rescale testing image
        //Image Data Generator  generates batches of tensor image data with real-time data
        //augmentation.
  Generate training set, testing set, and validation set
 Resize image to 32X32 for faster processing
 Step 2. Read the dataset
       dataset ← read resized image dataset
       label ←and label the data as: 0, 1, 2
    Here,
        0 is for Healthy
    1 is for Late_blight and
        2 is for Early_blight
Step 3. Create labeled dataset
        ldata ← [dataset, label] #combines the data and list to have a labeled data.
Step 4. Convert data to array
Step 5. Reshape the data to make it even.
Step 6. Divide by 255 to ensure that pixels are of same data type.
Step 7. Get the number of clusters
  import k-means modules and other required modules.
        Create a k-means object and fit the re-shaped data to predict.
        Read re_shaped data and append labels to it.
Step 8. Clean data
        Drop any Nan value from the dataset
        Save the clean file as potato.csv
Step 9.  Create a spark session.
Step 10. Read the csv using spark
Step 11. Print the schema to understand the structure of the dataset
Step 12. Drop any Nan value if present
Step 13. Apply PCA
        df ←Vectorize the labels
    df ←Extract features
    df ←Using vector assembler generate features
  assembled_data ← transform(df)
  scale ← using Standard Scaler generate standardized data
  data_scale ← fit assembled_data
  data_scale_output ← transform assembled_data
  pca ← apply PCA on data_scale_output
    Here,
      K ←3
      Input ← features
      Ouput ←pca features
Step 14. Apply K-means on results obtained after PCA
  scale ← generate standardized output usingfStandardScaler
  Here,
    input ←pca features
    output ← standardized
        data_scale ← fit scale to pca
        data_sclae_output ← transform pca using data_scale  object
Step 15. Create an object of evaluator
Step 16. Create a kmeans object
  fcol ← standardized
  ncluster ←10
    KMeans_Obj ←KMeans(featuresCol ←fcol, number of clusters ←ncluster)
         KMeans_fit ←fit KMeans_Obj on data_scale_output
Step 17. Evaluate the silhouette score
       silhouette coefficient, alternatively referred to as the silhouette score, is a metric used to determine the
goodness of a clustering algorithm. Its value is between -1 to 1.
     1: Indicates that clusters are clearly separated and distinct from one another.
     0: There is no substantial distance between clusters.
    -1: Indicates that clusters have been assigned incorrectly.
```

The spark environment is set using Python's PySpark module. Reading image data in a Pyspark environment is little different. The output is a PySpark data frame not the usual data frame as in case of Pandas. This implies that only commands from PySpark module can be used on the output data frame. The challenge with image classification is the speed. Reading all the images and classifying them is time consuming. We merged training, and testing images in one folder. Parallel processing or Pyspark helps us in reading the images into a data frame amazingly fast. Address to the problem of slow classification we first resized the images to 32X32. This helped in bringing a uniformity to the data as well. Further, we divided each pixel by 255 to get the pixel values between 0 and 1. Lower pixel values increases the speed of classification. The next task is to predict the disease.

2. Algorithm : Prediction

```
Step 1. Leaf ← upload a new leaf
Step 2. PcaNew ← extract features Leaf using PCA
Step 3. POutput ←KMeans_fit(PcaNEw) // Kmeans_fit is taken from algorithm 3.2.1
Step 4. If POutput==0:
Step 5.Disease ← Healthy
Step 6. else if POutput ==1:
Step 7.     Disease ←Late_blight
Step 8. else:
Step 9.Disease ← Early_blight
```

For predicting the disease, the object of the compiled model is taken. The extracted features of the new image are passed through the classification output to get the predicted label.

IV. Results

MCIP was evaluated on two datasets to ascertain the claim that the proposed work is independent of any dataset. The results are compared with the results obtained by existing techniques. We have designed three models to predict the plant diseases.

```
|pcaFeatures                                                  |
+-------------------------------------------------------------+
|[-0.004622912089267 4475,-12.19086263876783,3.2862372674221696]|
|[-1.0094535660927362,-12.954875385375228,1.7674674730146027]  |
|[-2.0043414488563416,-10.133650175268926,3.739270402051278]   |
|[-3.003278795586943,-11.718863322520273,3.648182581156293]    |
|[-4.014468615623557,-13.34162299930827,1.9707816854417384]    |
|[-5.01337073348173,-12.750845083798046,3.8028709339343023]    |
|[-6.015673025640713,-12.264992205955696,4.815775330746212]    |
|[-6.99866775467144,-13.193048776585503,1.5520065839634416]    |
|[-8.00910364346557,-13.433031151215603,1.546739370187543]     |
|[-9.006220901935423,-11.098952671903572,1.727128255097457]    |
|[-10.007918926096849,-9.938324376883212,0.2753788509600351]   |
|[-11.001068444602414,-11.578982174745462,2.279263303723642]   |
|[-12.00423762305103,-11.691066339815265,3.3854175258394137]   |
|[-13.006565656316202,-13.638162658248492,2.196565122769067]   |
|[-14.009194043859427,-12.17208420960617,0.9122012428916354]   |
|[-15.018159272444452,-11.624971710382743,3.811355796443408]   |
|[-16.013419850680407,-11.846870985673839,2.6789184271079707]  |
|[-17.004374168332184,-11.503518984374487,2.6345074834385245]  |
|[-18.00281313521687,-11.820212052906973,1.156557310881946]    |
|[-19.008819207011133,-13.843946605671729,1.7222872337880752]  |
+-------------------------------------------------------------
```

Fig.3 PCA feature extraction

Fig.3 is the output of the PCA features extract from the images. First 20 results are reproduced here. The data is of the first 20 leaves in the image dataset.

```
+--------------------+--------------------+
|         pcaFeatures|        standardized|
+--------------------+--------------------+
|[-0.0046229120892...|[-5.3287955980626...|
|[-1.0094535660927...|[-0.0116358944655...|
|[-2.0043414488563...|[-0.0231038914074...|
```

Fig. 4 standardized features

After PCA feature extraction, the data is standardized. The standardized features are shown in fig. 4. PCA and standardized features are used with k-means to classify the dataset. The outputs are of potato dataset only.

Table 6: Silhouette score using only K-means (potato)

Value of K	Score
1	0.7966620104390114
2	0.7410152961759711
3	0.7764226529818571
4	0.7556580195635196
5	0.74193107935930953
6	0.7720033864837685
7	0.77958633539164464
8	0.7409168118624311
9	0.6966620104390114

Table 6 is the output received for different values of silhouette for different values of K. The optimum value is with K=7. The accuracy achieved is 98%.

Table 7: Silhouette score MCIP (potato)

Value of K	Score
1	0.8966620104390114
2	0.8410152961759711
3	0.9886226529818571
4	0.8556580195635196
5	0.84193107935930953
6	0.8720033864837685
7	0.87958633539164464
8	0.8409168118624311
9	0.7966620104390114

Table 8 is the score received when we used PCA and K-means. MCIP performs best at K=3. The accuracy achieved is close to 100%.

Table 8: Outcome of classifying potato dataset

Model	TP	FP	TN	FN
[45]	4875	10	40	35
Deep Learning	4886	9	36	29
K-means	4852	15	44	39
MCIP	4961	0	1	0

True positive is kept at 100, which is the number of images. All the input images are known to be correct, there are no images which are detected but are not correct, thus FP is 0. The number of images which are Falsely detected negative are very low.

For comparison, we designed three models: Deep Learning, K-means, K-means + PCA (MCIP). The other comparison is done with [45] for potato dataset and [46] for rice dataset.

The deep learning model used has 6 Convolution 2D layers, 6 Maxpooling layers, 2 dropout layers, and two dense layer. The activation function used is Relu. The activation function used on the output dense layer is softmax. All the three models run on PySpark. K-means (without PCA), re-scales the images to 32X32 as the feature input. The value of k is again 3. We wanted to see if PCA would make any difference to classification and prediction. The results show that although k-means can give good results but MCIP gives much better results.

Table 9: Accuracy using Table 8

Model	Precision	Recall	Accuracy	F1-score
[45]	0.9928717	0.8	0.9909274	0.8860616
Deep Learning	0.9940997	0.8	0.9923387	0.8865502
K-means	0.9920262	0.7457627	0.9890909	0.8514453
MCIP	1	1	1	1

The accuracy of MCIP is close to 100%. We can safely say that the accuracy would range between 99.5 and 100%. The algorithm was fixed on a random state to reproduce the same results every time. The results may vary if the state changes. The fall in accuracy of [45] is due to higher number of images in MCIP.

Table 10: Outcome of classifying rice dataset

Model	TP	FP	TN	FN
[46]	5726	43	92	71
Deep Learning	5854	11	34	33
K-means	5793	31	43	65
MCIP	5929	0	2	1

Table 11: Accuracy using Table 10

Model	Precision	Recall	Accuracy	F1-score
[46]	0.9877523	0.6814815	0.9807822	0.8065196
Deep Learning	0.9943944	0.7555556	0.9925826	0.8586762
K-means	0.9889041	0.5810811	0.9838166	0.7320241
MCIP	0.9998314	1	0.9998314	0.9999157

The accuracy of deep learning and MCIP is 99.983% again close to 100%. Without any changes to the algorithm with Rice dataset also we get the similar results. The model can take any image classify it and predict the disease.

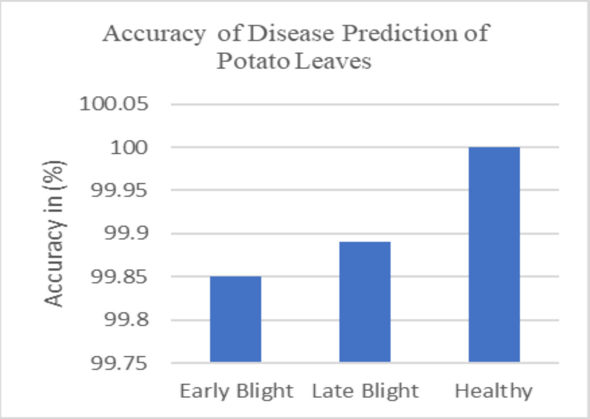


Fig.5 Disease Classification accuracy of Potato Leaf Dataset

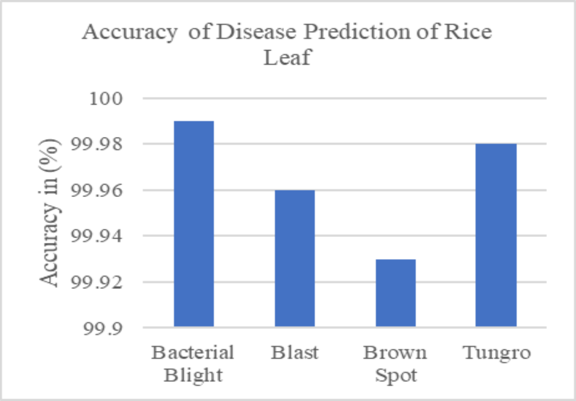


Fig.6 Disease Classification Accuracy Rice Dataset

Fig. 5 and Fig. 6 are plotted to show the accuracy of the predicted diseases. The accuracy varies for different diseases. The average classification accuracy is mentioned in table 9 and 11.

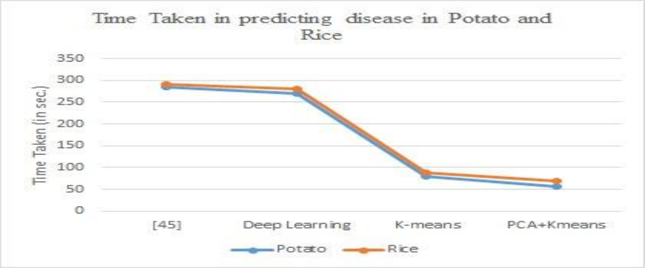


Fig. 7 Time taken by Potato and Rice Dataset

Time plays an important role in classification and prediction of an image dataset. We could reduce the time by 80% for potato dataset and 76% for rice leaf dataset. Paper (Rashid J et al. 2021) is on Potato leaf, we implemented the model on rice leaf to arrive at the results.

V. Conclusion

Deep learning techniques work significantly well on a non-image dataset. With images comes the constraint of time. According to our study many researchers have used image processing techniques like segmentation. Segmentation requires image to be dilated first and then edges are marked and finally they can be segmented. One can use Region of interest also to segment the image. All the techniques are time consuming when it comes to a large dataset. MCIP overcomes the challenge by creating a PySpark environment. We were able to significantly reduce the processing time between 80 and 76%. The accuracy achieved is close to 100% but that is not of much relevance as many techniques are close to 100%. MCIP gets an advantage over the other techniques due to its speed and ability to classify and predict any image dataset.

We faced a challenge to read images in Pyspark from folders. The benchmark datasets are designed for deep learning techniques which use Training, testing, and Validation set. Also, the read images are in Pyspark data frame. We could mitigate the issues after hit and trial technique. In future we would like to use PCA with Long Short-Term Memory (LSTM) to take the advantage of forget gate. We would continue to use PySpark environment to maintain the speed.

Author’s Contribution

- As per the literature survey and our knowledge this is the first attempt of classifying and predicting leaf disease on Pyspark.
- We have reduced the time drastically.
- MCIP can classify any image dataset
- MCIP can predict any leaf disease.

Declarations

Conflict of interest: No potential conflict of interest is reported by authors.

References

[1]Velusamy, P., Rajendran, S., Mahendran, R. K., Naseer, S., Shafiq, M., & Choi, J. G. (2021). Unmanned Aerial Vehicles (UAV) in precision agriculture: applications and challenges. *Energies*, *15*(1), 217.

[2]Dhaliwal, G. S., Jindal, V., & Dhawan, A. K. (2010). Insect pest problems and crop losses: changing trends. Indian Journal of Ecology, 37(1), 1-7.

[3]Hiremath, S., Wittke, S., Palosuo, T., Kaivosoja, J., Tao, F., Proll, M., ... & Mamitsuka, H. (2021). Crop loss identification at field parcel scale using satellite remote sensing and machine learning. *PloS one*, *16*(12), e0251952.

[4]Muimba-Kankolongo, A. (2018). *Food Crop Production by Smallholder Farmers in Southern Africa: Challenges and Opportunities for Improvement*. Academic Press.

[5]Das, I. K., & Rajendrakumar, P. (2016). Disease resistance in sorghum. In *Biotic stress resistance in millets* (pp. 23-67). Academic Press.

[6]Sasaki, Y., Okamoto, T., Imou, K., & Torii, T. (1998). Automatic diagnosis of plant disease-Spectral reflectance of healthy and diseased leaves. *IFAC Proceedings Volumes*, *31*(5), 145-150.

[7]Henson, J. M., & French, R. (1993). The polymerase chain reaction and plant disease diagnosis. *Annual review of phytopathology*, *31*(1), 81-109.

[8]Koo, C., Malapi-Wight, M., Kim, H. S., Cifci, O. S., Vaughn-Diaz, V. L., Ma, B., ... & Han, A. (2013). Development of a real-time microchip PCR system for portable plant disease diagnosis. *PloS one*, *8*(12), e82704.

[9]Prasad, S., Peddoju, S. K., & Ghosh, D. (2016). Multi-resolution mobile vision system for plant leaf disease diagnosis. *Signal, image and video processing*, *10*(2), 379-388.

[10]Ali, H., Lali, M. I., Nawaz, M. Z., Sharif, M., & Saleem, B. A. (2017). Symptom based automated detection of citrus diseases using color histogram and textural descriptors. *Computers and Electronics in agriculture*, *138*, 92-104.

[11]Singh, U. P., Chouhan, S. S., Jain, S., & Jain, S. (2019). Multilayer convolution neural network for the classification of mango leaves infected by anthracnose disease. *IEEE Access*, *7*, 43721-43729.

[12]Zhang, K., Xu, Z., Dong, S., Cen, C., & Wu, Q. (2019). Identification of peach leaf disease infected by Xanthomonas campestris with deep learning. *Engineering in Agriculture, Environment and Food*, *12*(4), 388-396.

[13]Lu, J., Hu, J., Zhao, G., Mei, F., & Zhang, C. (2017). An in-field automatic wheat disease diagnosis system. *Computers and electronics in agriculture*, *142*, 369-379.

[14]Deepa, N. R., & Nagarajan, N. (2021). Kuan noise filter with Hough transformation based reweighted linear program boost classification for plant leaf disease detection. *Journal of Ambient Intelligence and Humanized Computing*, *12*(6), 5979-5992.

[15]Hamuda, E., Mc Ginley, B., Glavin, M., & Jones, E. (2018). Improved image processing-based crop detection using Kalman filtering and the Hungarian algorithm. *Computers and electronics in agriculture*, *148*, 37-44.

[16]Mahmud, M., Kaiser, M. S., Hussain, A., & Vassanelli, S. (2018). Applications of deep learning and reinforcement learning to biological data. *IEEE transactions on neural networks and learning systems*, *29*(6), 2063-2079.

[17]Picon, A., Alvarez-Gila, A., Seitz, M., Ortiz-Barredo, A., Echazarra, J., & Johannes, A. (2019). Deep convolutional neural networks for mobile capture device-based crop disease classification in the wild. *Computers and Electronics in Agriculture*, *161*, 280-290.

[18]Ferentinos KP. (2018)Deep learning models for plant disease detection and diagnosis. Computers and electronics in agriculture,145(1),311-318.

[19]Huang, T., Yang, R., Huang, W., Huang, Y., & Qiao, X. (2018). Detecting sugarcane borer diseases using support vector machine. *Information processing in agriculture*, *5*(1), 74-82.

[20]Singh, U. P., Chouhan, S. S., Jain, S., & Jain, S. (2019). Multilayer convolution neural network for the classification of mango leaves infected by anthracnose disease. *IEEE Access*, *7*, 43721-43729.

[21]Singh, V. (2019). Sunflower leaf diseases detection using image segmentation based on particle swarm optimization. *Artificial Intelligence in Agriculture*, *3*, 62-68.

[22]Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., & Stefanovic, D. (2016). Deep neural networks based recognition of plant diseases by leaf image classification. *Computational intelligence and neuroscience*, *2016*.

[23]Sharma, P., Berwal, Y. P. S., & Ghai, W. (2020). Performance analysis of deep learning CNN models for disease detection in plants using image segmentation. *Information Processing in Agriculture*, *7*(4), 566-574.

[24]Agarwal, M., Singh, A., Arjaria, S., Sinha, A., & Gupta, S. (2020). ToLeD: Tomato leaf disease detection using convolution neural network. *Procedia Computer Science*, *167*, 293-301.

[25]Mishra, S., Sachan, R., & Rajpal, D. (2020). Deep convolutional neural network based detection system for real-time corn plant disease recognition. *Procedia Computer Science*, *167*, 2003-2010.

[26]Khamparia, A., Saini, G., Gupta, D., Khanna, A., Tiwari, S., & de Albuquerque, V. H. C. (2020). Seasonal crops disease prediction and classification using deep convolutional encoder network. *Circuits, Systems, and Signal Processing*, *39*(2), 818-836.

[27]Hussain, A., Ahmad, M., Mughal, I. A., & Ali, H. (2018). Automatic disease detection in wheat crop using convolution neural network. In *The 4th International Conference on Next Generation Computing*.

[28]Venkataramanan A, Honakeri DK, Agarwal P.(2019).Plant disease detection and classification using deep neural networks. Int. J. Comput. Sci. Eng. 11(9):40-6.

[29] Chandy, A. (2019). Pest infestation identification in coconut trees using deep learning. *Journal of Artificial Intelligence*, *1*(01), 10-18.

[30]Karthik, R., Hariharan, M., Anand, S., Mathikshara, P., Johnson, A., & Menaka, R. (2020). Attention embedded residual CNN for disease detection in tomato leaves. *Applied Soft Computing*, *86*, 105933.

[31]Zhang, Y., Song, C., & Zhang, D. (2020). Deep learning-based object detection improvement for tomato disease. *IEEE Access*, *8*, 56607-56614.

[32]Tarik MI, Akter S, Al Mamun A, Sattar A. Potato Disease Detection Using Machine Learning (2021). Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV),IEEE, 800-803.

[33]Nazki, H., Yoon, S., Fuentes, A., & Park, D. S. (2020). Unsupervised image translation using adversarial networks for improved plant disease recognition. *Computers and Electronics in Agriculture*, *168*, 105117.

[34]Ganatra, N., & Patel, A. (2020). Performance Analysis Of Fine-Tuned Convolutional Neural Network Models For Plant Disease Classification. *Published by International Journal of Control and Automation*, *13*(3), 293-305.

[35]Sambasivam, G., & Opiyo, G. D. (2021). A predictive machine learning application in agriculture: Cassava disease detection and classification with imbalanced dataset using convolutional neural networks. *Egyptian Informatics Journal*, *22*(1), 27-34..

[36]Geetharamani, G., & Pandian, A. (2019). Identification of plant leaf diseases using a nine-layer deep convolutional neural network. *Computers & Electrical Engineering*, *76*, 323-338.

[37]Liang, Q., Xiang, S., Hu, Y., Coppola, G., Zhang, D., & Sun, W. (2019). PD2SE-Net: Computer-assisted plant disease diagnosis and severity estimation network. *Computers and electronics in agriculture*, *157*, 518-529.

[38]Khalifa, N. E. M., Taha, M. H. N., El-Maged, A., Lobna, M., & Hassanien, A. E. (2021). Artificial Intelligence in Potato Leaf Disease Classification: A Deep Learning Approach. In *Machine Learning and Big Data Analytics Paradigms: Analysis, Applications and Challenges* (pp. 63-79). Springer, Cham.

[39]Rozaqi, A. J., & Sunyoto, A. (2020, November). Identification of Disease in Potato Leaves Using Convolutional Neural Network (CNN) Algorithm. In *2020 3rd International Conference on Information and Communications Technology (ICOIACT)* (pp. 72-76). IEEE.

[40]Sanjeev, K., Gupta, N. K., Jeberson, W., & Paswan, S. (2021). Early Prediction of Potato Leaf Diseases Using ANN Classifier. *Oriental Journal of Computer Science and Technology*, *13*(2, 3), 129-134.

[41]Barman, U., Sahu, D., Barman, G. G., & Das, J. (2020, July). Comparative Assessment of Deep Learning to Detect the Leaf Diseases of Potato based on Data Augmentation. In *2020 International Conference on Computational Performance Evaluation (ComPE)* (pp. 682-687). IEEE.

[42]Johnson, J., Sharma, G., Srinivasan, S., Masakapalli, S. K., Sharma, S., Sharma, J., & Dua, V. K. (2021). Enhanced field-based detection of potato blight in complex backgrounds using deep learning. *Plant Phenomics*, *2021*.

[43]Lee, T. Y., Yu, J. Y., Chang, Y. C., & Yang, J. M. (2020, February). Health detection for potato leaf with convolutional neural network. In *2020 Indo–Taiwan 2nd International Conference on Computing, Analytics and Networks (Indo-Taiwan ICAN)* (pp. 289-293). IEEE.

[44]Islam, M., Dinh, A., Wahid, K., & Bhowmik, P. (2017, April). Detection of potato diseases using image segmentation and multiclass support vector machine. In *2017 IEEE 30th canadian conference on electrical and computer engineering (CCECE)* (pp. 1-4). IEEE.

[45]Rashid, J., Khan, I., Ali, G., Almotiri, S. H., AlGhamdi, M. A., & Masood, K. (2021). Multi-Level Deep Learning Model for Potato Leaf Disease Recognition. *Electronics*, *10*(17), 2064.

[46]Sethy, P. K., Barpanda, N. K., Rath, A. K., & Behera, S. K. (2020). Deep feature based rice leaf disease identification using support vector machine. *Computers and Electronics in Agriculture*, *175*, 105527.

[47]Jonathan, F., Yang, D., Gowing, G., & Wei, S. (2021, December). Machine Learning Framework for Detecting Offensive Swahili Messages in Social Networks with Apache Spark Implementation. In *2021 IEEE International Conference on Progress in Informatics and Computing (PIC)* (pp. 293-297). IEEE.

[48]Dutta, P., Shah, N., & Saha, S. (2021, October). A Multi-Objective Optimization-based Clustering Approach for CORD-19 Scholarly Articles. In *2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC)* (pp. 1393-1398). IEEE.