PataNET: A Convolutional Neural Networks to Identify Plant from Leaf Images

Md. Majedul Islam
Dept. of Computer Science & Engineering
Daffodil International University
Dhaka, Bangladesh
majedul15-6784@diu.edu.bd

Md. Hafizur Rahman Arfin
Dept. of Computer Science & Engineering
Daffodil International University
Dhaka, Bangladesh
rahman15-7144@diu.edu.bd

AKM Shahahriar Azad Rabby
Dept. of Computer Science & Engineering
Daffodil International University
Dhaka, Bangladesh
azad15-5424@diu.edu.bd

Syed Akhter Hossain
Dept. of Computer Science & Engineering
Daffodil International University
Dhaka, Bangladesh
aktarhossain@daffodilvarsity.edu.bd

***Abstract*—Plants are everywhere around the Earth. Plant identification is a very important problem for environmental protection and exploration. But getting to know various plants requires taking into account a large number of features which might be easy for a botanist but not easy for ordinary people. So having an automatic Plant identifier which uses a leaf to identifying plants will help many people. Here, a method is proposed where Convolutional Neural Network (CNN) technique is used to classify plants using leaf images. Using Adam optimizer and automatic Learning Rate reduction technique the model gave promising accuracy. This system was trained on 3600 RGB leaf images of 2 categories for 6 different plant species. The model reported promising results with validation accuracy was 95.86% and training accuracy was 96.54%. Different pre-processing techniques such as background whitening, noise removal are used. In the convolutional networks activation function ReLU is used in the hidden layer and Softmax for output layer.**

***Index Terms*—Convolutional neural network, Leaf classification, Adam optimizer, automatic Learning Rate Reduction.**

I. Introduction

Plants have very deep ecological relationship with animal life. They are the most vital part of nature. Recognizing plants through various methods has always been a field of interest for many researchers. Plants identification knowledge are used in medical science, food production, in education, etc. There are many reasons behind why plant recognition is important.

Plant identification is a very relevant and essential problem to solve in the current time period. Many work is happening recently to recognize plant disease from its leaf [1] [2]. But not many focused in plant detection. It is still difficult to recognize a plant for an ordinary person. Gathering information of a plant from the local library or going to the nursery is time-consuming. Recognizing plant requires the expertise of botanist. Understanding different geometrical and morphological features of plant is hard. Physiological features like length, width, diameter, parameter, area, aspect ratio, shape, size, color of different plants are very different from one another. There are more than 391,000 vascular plant species and more than 369,000 flower species in the world. Getting little features right and comparing them with other similar featured can cause errors in identifying plant. It is also a lengthy and tedious process.

In this 21st century due to advancement in computer science and technology, people are relying on their phones and computers to automate and solve most of their problems. And recognizing plants using images can be automated. So that both people who are working as a botanist and who are from another field can have an edge on detecting plants. An image-based detection is faster, easier and less prone to error. It will facilitate a lot of people and speed up the plant species identification process.

A leaf is an organ of a plant. Using leaf images to identify a plant can help to solve this problem. Leaves are abundant in a tree. Other than winter, It is easier to get a leaf image than a fruit or a flower image. Also, the inequality in the characterizes of root and steam are hard to characterize. Leaves have enough characterization features on them to differentiate different species. Identifying plant by leaf images is feasible than other ways of identification. This is why among the other segments of trees, the leaf has attracted the most attention of the researchers for detecting plant through images.

Plant recognition through leaf images is still a very difficult task due to inefficient models and improper representation approaches. Also, the subtlety of leaf characteristics from plant to plant due to phenotypic plasticity makes the problem complex. The previous solution to this problem used different classification techniques that required predefined feature and a complex preprocessing of images. Recently the improvement in Convolutional Neural Network(CNN) [3] and advancement in computer vision technology has made things easier. From object recognition from images [4] to speech recognition from audio [5] CNN has showed remarkable results. Back-propagation algorithm was integrated in 1988 [6]. Now CNN has revolutionized unsupervised learning by identifying patterns in images and classifying them without the need of a lot of feature definition and pre-processing. Though it was not specifically made for images it has achieved very good results in problems like recognizing images and classifying them.

In this paper, a model is proposed based on Convolutional Neural Network(CNN) to identify plants from pictures. The model reported validation accuracy 95.86% and training accuracy 96.54%. The training and development time was significantly lesser than previous solutions of this problem. 3600 images of 6 species of plants were used to train the model. No public dataset was used to train the model. Because public datasets like Flavia dataset, Pl@ntLeaves II has more species but images for single species are not enough. CNN model needs a lot of images to achieve desired accuracy. So we decided to make a dataset with more images of a single plant species. We collected data for 6 species, total 600 for each species.

II. Literature Review

In several works of literature, many plant recognition systems have been successfully developed for leaf detection using both traditional Neural Networks and Convolutional Neural Networks.

For instance Luciano D. S. Pacifico et al. [7] proposed that, for traditional artificial neural network approaches to recognize plant from different set of features, Multi-Layer Perceptron with Backpropagation algorithm performed better than K-Nearest Neighbors (KNN), Decision Tree classifier (DT), Naive Bayes Classifier (NB) and Support Vector Machine (SVM). The MLP training time was higher than the other algorithms but the accuracy was also better than other algorithms. The algorithm was tested on 3 datasets collected from UCI Machine Learning Repository. Datasets were Iris, Seeds and 100 Plant Leaves. The important emphasis was on feature extraction of plant leaves dataset, without getting the right features the algorithms could not give better accuracy and even selecting wrong features might degrade the performance.

In another work Aparajita Sahay et al. [8] Presented a system for leaf identification system that had 3 steps, First is pre-processing step, The noise from the images was removed. They used gray images instead of green for recognizing patterns better and remove the problem with the color variation of same plant images. Second is feature extraction, the SIFT or scale-invariant feature transform was used to get the features. Third is the proposed model, the proposed model used weighted KNN. Precision and Recall measurement metrics were used to measure accuracy. For precision result, the normal KNN gave 66.3% accuracy, weighted KNN gave 85.2% accuracy. For recall result, normal KNN gave 50.9% accuracy and weighted KNN gave 67.5%. The dataset used was LeafSnap. The accuracy was tested in three of data category, Intra-family, and Inter-family and Mixed. For Mixed, training data was 200 and testing data was 100. In another paper VIJAY SATTI et al. [9] compared the performance of two algorithms ANN and KNN on images of leaves to identify a plant. The accuracy results were, for ANN is 93.3% and for KNN is 85.9% they have performed pre-processing on the images and extracted features from the images. The dataset used Flavia dataset which has 1907 sample leaves of 33 different species. The amount of data are significantly less for evaluating the model's performance. But the comparison on ANN and KNN resulted in ANN being most accurate and scalable than KNN.

In works with Convolutional Neural Nets Vladimir V. Mokeev [10] in his recent paper on plant classification using CNN proposed a model with an architecture of 4 convolutional layers, 4 dropout layers, 4 polling layers and fully connected layers at the end. Among different models that are tested, the best-performed model in this paper resulted in an F1 score of 94.63% in 55 epoch. Out of 950 test images, this architecture correctly identified 906 images. Different learning rate modification techniques were used but the best performed was drop based learning rate. But the learning rate was pre-determined to reduce after a certain number of epoch, regardless of the improvement of accuracy. Dataset had a total 4750 images of 12 species. Image augmentation techniques like rotation, flipping, cropping, zooming, blurring, contrasting, etc. were used to reduce overfitting. Other models like Random forest, Extremely Randomized Tree, Light Gradient Boosting and CNN model could not exceed 60% score in this dataset.

Xiang He et al. [11] in his paper proposed a method for leaf classification by improved CNN and Single Connected Layer(SCL). The Dropout technique and Gaussian filter were used to reduce overfitting chances. The experiment was done on ICL leaf image database. 500 data for training and 100 data for testing. 20 kinds of species used. 4 models were tested, among them basic CNN accuracy was 82.0%, CNN with SCL was 87.7% where CNN and improved SCL gave 91.9% accuracy. In another paper Yan-Hao Wu et al. [12] in their paper used Convolutional Neural Network to classify leaf images. A tweaked reduced version of AlexNet was used for the model and PReLU was used as activation function instead of ReLU. This paper claims that reduction of AlexNet and using PReLU instead of ReLU can have 10% increase in performance. This model used RGB as an input for the model instead of a GrayScale image. This system was trained by their own dataset of 1500 leaves of 50 kinds of plants. The accuracy achieved was 94.8%.

Zhong-Qiu Zhao et al. [13] in his paper used Growing Convolutional Neural Network with Progressive Simple Learning Algorithm. GCNN is different from CNN because it grows to suit the classification task. This paper claims that GCNN performs better than traditional CNN and SVM classifiers. Progressive Simple Learning Algorithm automatically chooses the number of training samples avoiding under-learning or over-fitting. They used ImageCLEF2012 dataset to measure the accuracy of the model. The accuracy of the model was 88.14% with GCNN (9 branches)+PLSM.

From this literature survey, it has been observed that the traditional Neural Network method approach needs very complex preprocessing and feature extraction techniques before training the model. Selecting the correct feature and training large image dataset is hard in ANN or KNN solutions. Also the features extraction and solutions very much are task dependent and cannot be generalized. Which binds us from taking the work outside of the bound of particular dataset. It is also time consuming and not as efficient as CNN based approach. CNN approach can solve the problem with feature extraction .Convolutional Neural Networks shows promising results without putting much effort into feature extractions. Previous work on the CNN models are trained on small datasets. Also The accuracy of the previous works can be improved. The accuracy of CNN can be improved with the right choice of optimizer and hyperparameters. Proper comparisons with other models is missing in some of the previous work. Which is necessary for better understanding the improvement. We propose a model that can achieve better accuracy also compared it with other models. The following section is to describe the work and model.

III. Proposed Methodology

The proposed method phases like Data Acquisition, Dataset preparation, Proposed model, etc. All of which are described below.

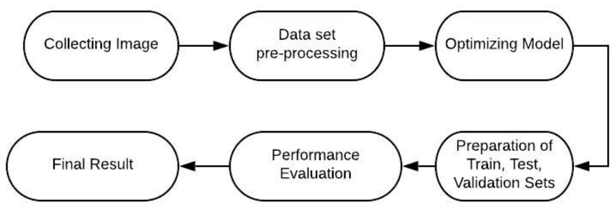

Fig. 1: Work flow

A. Data Acquisition

The images of leaves were collected using a digital camera. Since the camera did not have lossless compression format other than JPEG, the images were saved as JPEG 3 channel image format. The dataset contains images of leaves of 6 species. Each species contains images of 2 categories. Category-1 images have a single leaf on a white background, and Category-2 images were directly taken from the tree, they have natural leaves and trunks as background behind them. Each species has 500 images of Category-1 and 100 images of Category-2. Most of the images have no noise and the labeling is correct. Joining Category-1 and Category-2 images made a dataset that has (500+100)*6 = 3,600 images of leaves of 6 species. 6 classes in the dataset are shown in figure-2.

B. Dataset Preparation

Some Category-1 images did not have a fully white background. So they were manually whitened to fit the dataset. It is easier for the CNN model to work with square images. So the initial images were cropped from wide to square. Images of the dataset were Full HD, meaning they have resolution over 1920 x 1080 pixel. Training the model with high-resolution image is computationally expensive so the images were resized to

(a) Green-orange leaf (b) Guava leaf (c) Jackfruit leaf (d) Lichi leaf

(f) Sugar-apple leaf (e) Mango leaf

Fig. 2: Dataset example

128 x 128 pixel. Also, to reduce the illumination differences, Minmax(1) normalization was used on the dataset.

$$Z_i = \frac{X_i - minimum(X)}{maximum(X) - minimum(X)} \tag{1}$$

C. Proposed Model

Convolutional Neural Network (CNN) are getting very popular and showing promising results in the field of computer vision [14], [15]. The main reason behind its widespread use is its efficiency in pattern classification. It can give better performance than other long-established methods [16].

CNN layers are made of two components. The are convolutional layers and max-pooling layers. Convolutional layers have filters that take a small local part of the image and processes it and it is replicated on the whole image. Max-pooling layers take maximum filter activations from the filtered image and a convolutional layer of lower resolution is created. This process adds tolerance and translation invariance to little differences of positions in objects parts.

This is a multilayered CNN model. The proposed model uses optimizer named ADAM [17]. The initial learning rate was set to 0.001. Which was automatically reduced as the learning rate progressed. For convolution 1, filter = 64, kernel size = (3 × 3), stride = (1 × 1), padding is set to “same” and activation function is ReLU (2). After that a (2 × 2) max-pooling layer with stride (2 x 2) was set.

$$ReLU(X) = MAX(0, X) \tag{2}$$

For convolution layer 2 and convolution layer 3, the filter = 64, kernel size = (3 × 3), Stride = (1 × 1), “same” padding with activation function ReLU is used. In between convolution layer 2 and convolution layer 3 there was a (2 x 2) max-pooling layer with stride = (2 x 2) was set. For convolution layer 4 the filter = 96, kernel size = (3 × 3), stride = (1 × 1), “same” padding with activation ReLU (2). Then a (2 × 2) max-pooling layer with stride = (2 x 2) is placed.

In the last convolution layer, convolution 5, filter = 192, kernel size = (3 × 3), stride = (1 × 1), “same” padding with ReLU activation function. Followed by a (2 × 2) max-pooling layer with stride = (2 x 2). A 25% dropout is used [18] to reduce overfitting.

Then flattened the image and a dense layer with 128 units was placed. The activation function was ReLU. 50% dropout used to reduce overfitting at the final output layer, with using 6 units SoftMax (3) activation. The error metric Categorical Cross Entropy (4) was used. Figure 3 is showing the neural network architecture.

$$\sigma(z)_j = \frac{e^{z_i}}{\sum_{K=1}^{K} e^{z_k}} for j = 1, ...k \tag{3}$$

$$L_i = -\sum_j t_{i,j} \, log(p_i, j) \tag{4}$$

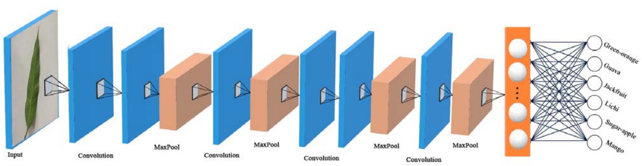

Fig. 3: Proposed Model

D. Optimization

The right choice for optimization algorithm in deep learning models can have noticeable improvement in both reduction in training time and improvement in accuracy. First published in 2014 ADAM is a is an optimization algorithm with adaptive-learning-rate specifically for training deep neural networks. Recently in deep learning applications like computer vision, ADAM optimizer is getting immense popularity. This algorithm combines the good parts of both Adaptive Gradient Algorithm (AdaGrad) [19] and Root Mean Square Propagation (RMSProp) [20]. ADAM sets individual learning rates at each parameter from approximations of first moment and second moments at the gradients. This algorithm is an update to traditional stochastic gradient descent algorithm. Instead of the classical stochastic gradient descent, ADAM optimizer was used for better performance.

Optimizer ADAM (5) with beginning learning rate = 0.001 was used in the proposed model.

$$v_t = (1 - \beta_2) \sum_{i=1}^{t} \beta_2^{t-i} . g_i^2 \tag{5}$$

Instead of traditional mean square loss, cross-entropy performs better in classification problem [21]. So instead of mean-square, cross-entropy error was calculated in the model. For cross entropy error is unlikely to stall out because of the weight changes. They do not get smaller and smaller as the training progresses.

Learning rate determines the steps by which the values of parameters will be updated. Too high or too low learning rate are not suitable for the optimizer. But getting that perfect learning rate intuitively not always gives the best result. In the proposed model automatic Learning Rate reduction technique [20] was used. The learning rate is changed dynamically by monitoring validation accuracy.

IV. Experiment and Results

The experiments and results are discussed below.

A. Model Training

The proposed model was trained on defined training set and validation set created from the dataset. Batch size was set to 30. The automatic Learning Rate Reduce algorithm monitored the accuracy. At first learning rate had to be initialized to a value but after that It was adjusted accordingly. After 50 epochs, the accuracy was monitored manually and the learning rate was reduced. The model was trained again for 5-10 epoch to achieve good accuracy.

B. Model Evaluation

The model proposed here was trained and tested on the dataset. The accuracy found was promising compared to other related works [22]. The results on the dataset are described in the followings.

C. Train, Test, Validation Sets

The leaf database was collected by us. Leaf samples are collected from specially selected 6 specific species of plants. The initial size of images was 4000x3000 pixel which was resized to 128x128 pixel. Resized 128x128 pixel images were not turned into grayscale. This model uses 3 channel image. Because the details are better represented in RGB. The image was cropped for resizing to maintain the square aspect ratio. To train the model with proper patterns and characteristics of leaves background of the images were cleared and painted white.

The proposed model was trained on 2 categories of images. Images were divided into 2 sets. Set 1 had 80% of data for training the model and Set 2 had 20% for testing the model. Each species of plant had 600 images. To train the model, a total of 500 images were used. And other 100 images were used to test the model. So we have 3000 images for the training set and 600 for testing.

D. Model Performance

Parameters that are not going to be learned in the training process but needed to be set are known as hyperparameters. Some of the hyperparameters are learning rate of the model, iteration/epoch, batch-size, etc. In the other related models, the hyperparameter learning rate is set manually, but here the automatic Learning Rate reduction technique was used. Other hyperparameters were set intuitively. The table (I) shows the hyperparameters for our proposed model.

After training the model for 10 epoch model gets validation accuracy 43.44%. The model ran for 50 epoch resulted training

TABLE I: Hyper Parameters

Learning Rate	Iteration	Batch-Size
Automatic Rate Reduction	50	30

TABLE II: Training-Test Results

Epochs	Loss	Training Accuracy	Validation Accuracy
10	0.5593	0.8013	0.4344
30	0.1829	0.9394	0.9220
40	0.1133	0.9596	0.9503
50	0.1001	0.9654	0.9586

accuracy 96.54% and validation accuracy 95.86%. At epoch 29 the learning rate was changed from 0.0005 to 0.0025. We can see improvement in accuracy 3% due to changing in learning rate. The learning rate is further reduced in iterations and the best resulted 96.54% training accuracy and 95.86% validation accuracy. Validation and training accuracy changes in different epochs are show in table (II)

The accuracy and the loss graph is shown in figure - 4 and in figure - 5. The confusion matrix is shown in figure - 6 The comparison between proposed model and other models are shown in table (III)

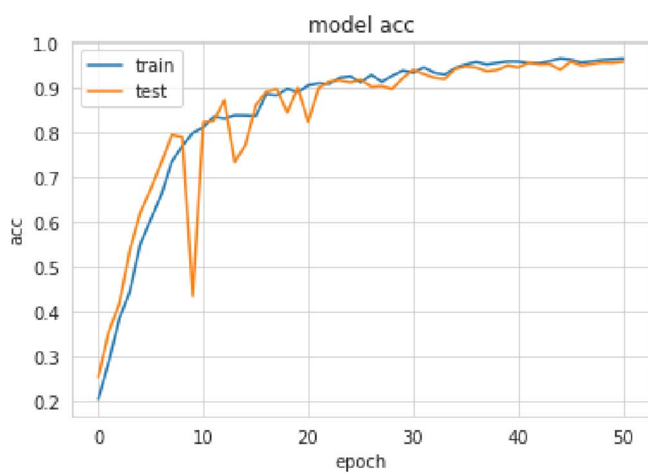

Fig. 4: Model Accuracy Evaluation

V. Conclusion

In this paper, an effective way of tackling the identification of plants from images of leaves by CNN and Adam Optimizer is presented. Using CNN with Adam optimizer and automatic Learning Rate reduction technique. The presented model gave

TABLE III: Model accuracy comparison

Work	Model	Accuracy	Dataset
Aparajita Sahay [8]	Weighted	85.2%	LeafSnap
VIJAY SATTI [9]	ANN	93.3%	Flavia
Vladimir V. [10]	CNN	94.63%	Plant Seedlings
Xiang He [11]	CNN+SCL	91.9%	ICL leaf dataset
Yan-Hao Wu [12]	CNN+AlexNet	94.8%	Private dataset
Zhong-Qiu Zhao [13]	GCNN+PLSM	88.14%	CLEF2012
Proposed model	CNN+Adam	96.54%	Private dataset

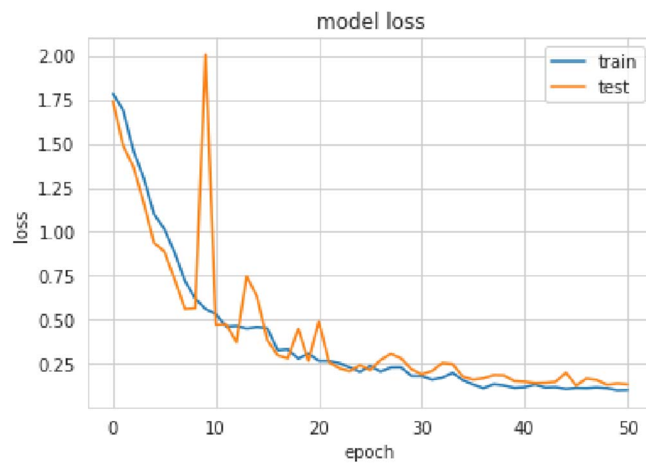

Fig. 5: Model Loss Evaluation

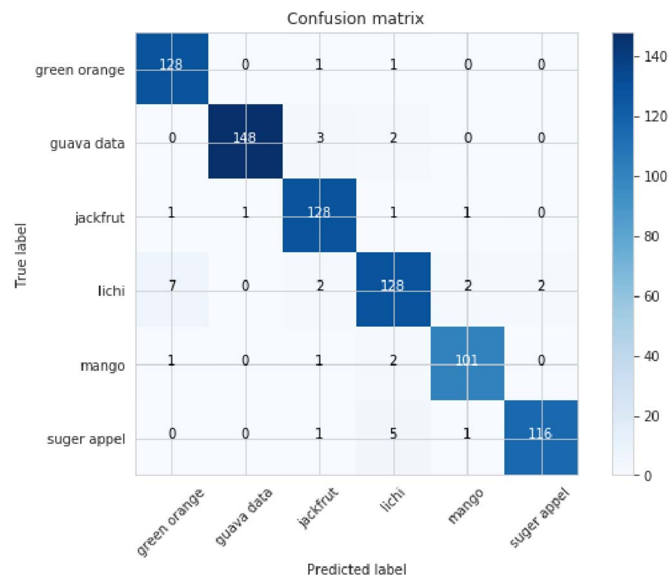

Fig. 6: Confusion Matrix

promising results on classification and identifying plants. The accuracy of the model was better compared to other types of related works. We used our own dataset to train the model. A total of 3600 data. The dataset contained significantly more data than other datasets for single plant.

VI. Future Work

Future work can include increasing the species in the dataset to test the model. Testing the model in other publicly available datasets and measuring accuracy. Also improving the model's performance on complex leaf images will be an extension of this work.

References

[1] J. Chen, Q. Liu, and L. Gao, "Visual tea leaf disease recognition using a convolutional neural network model," *Symmetry*, vol. 11, no. 3, p. 343, 2019.

[2] J. Amara, B. Bouaziz, A. Algergawy *et al.*, "A deep learning-based approach for banana leaf diseases classification." *BTW (Workshops)*, pp. 79–88, 2017.

[3] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.

[4] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[5] Y. LeCun, Y. Bengio *et al.*, "Convolutional networks for images, speech, and time series," *The handbook of brain theory and neural networks*, vol. 3361, no. 10, p. 1995, 1995.

[6] D. E. Rumelhart, G. E. Hinton, R. J. Williams *et al.*, "Learning representations by back-propagating errors," *Cognitive modeling*, vol. 5, no. 3, p. 1, 1988.

[7] L. D. Pacifico, V. Macario, and J. F. Oliveira, "Plant classification using artificial neural networks," *2018 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–6, 2018.

[8] A. Sahay and M. Chen, "Leaf analysis for plant recognition," *2016 7th IEEE International Conference on Software Engineering and Service Science (ICSESS)*, pp. 914–917, 2016.

[9] V. Satti, A. Satya, and S. Sharma, "An automatic leaf recognition system for plant identification using machine vision technology," *International journal of engineering science and technology*, vol. 5, no. 4, p. 874, 2013.

[10] V. V. Mokeev, "On application of convolutional neural network for classification of plant images," *2018 Global Smart Industry Conference (GloSIC)*, pp. 1–6, 2018.

[11] X. He, G. Wang, X.-P. Zhang, L. Shang, and Z.-K. Huang, "Leaf classification utilizing a convolutional neural network with a structure of single connected layer," *International Conference on Intelligent Computing*, pp. 332–340, 2016.

[12] Y.-H. Wu, L. Shang, Z.-K. Huang, G. Wang, and X.-P. Zhang, "Convolutional neural network application on leaf classification," *International Conference on Intelligent Computing*, pp. 12–17, 2016.

[13] Z.-Q. Zhao, B.-J. Xie, Y.-m. Cheung, and X. Wu, "Plant leaf identification via a growing convolution neural network with progressive sample learning," *Asian Conference on Computer Vision*, pp. 348–361, 2014.

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

[15] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[16] K. Chatfield, V. S. Lempitsky, A. Vedaldi, and A. Zisserman, "The devil is in the details: an evaluation of recent feature encoding methods." *BMVC*, vol. 2, no. 4, p. 8, 2011.

[17] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[18] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[19] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *Journal of Machine Learning Research*, vol. 12, no. Jul, pp. 2121–2159, 2011.

[20] T. Tieleman and G. Hinton, "Divide the gradient by a running average of its recent magnitude. coursera: Neural networks for machine learning," *Technical Report.*, 2017.

[21] K. Janocha and W. M. Czarnecki, "On loss functions for deep neural networks in classification," *arXiv preprint arXiv:1702.05659*, 2017.

[22] A. Sabu and K. Sreekumar, "Literature review of image features and classifiers used in leaf based plant recognition through image analysis approach," *2017 International Conference on Inventive Communication and Computational Technologies (ICICCT)*, pp. 145–149, 2017.