Bangladeshi Plant Recognition using Deep Learning based Leaf Classification

Conference Paper · July 2019

DOI: 10.1109/IC4ME247184.2019.9036515

Sultana Habiba
Khulna University of Engineering and Technology

Md. Khairul Islam
Khulna University of Engineering and Technology

Sk Md Masudul Ahsan
Khulna University of Engineering and Technology

International Conference on Computer, Communication, Chemical, Materials and Electronic Engineering (IC4ME2), 11-12 July, 2019

Bangladeshi Plant Recognition using Deep Learning based Leaf Classification

Sultana Umme Habiba, Md. Khairul Islam, Sk. Md. Masudul Ahsan
Department of Computer Science and Engineering
Khulna University of Engineering & Technology
Khulna - 9203, Bangladesh
suhabiba12@gmail.com, mdkislam27@gmail.com, smahsan@cse.kuet.ac.bd

***Abstract* - At present deep learning-based object recognition approaches have placed a tremendous effect for classifying different objects. Leaves recognition using supervised learning has shown satisfying performance which may help in various research purposes also. In our work, we have used a deep convolutional neural network as a classifier. We have used a transfer learning approach. We have prepared our work dataset based on Bangladeshi plants which contains eight different classes of leaves. We have experimented with VGG16, VGG19, Resnet50, InceptionV3, Inception-Resnetv2 and Xception deep convolutional neural network models where we have found the highest value in VGG 16 which shows almost 96% classification accuracy. Recognition of useful plants using leaf image will be greatly helpful in the research of ayurvedic and endangered plants.**

Keywords—leaf classification, deep convolutional neural network, transfer learning.

I. Introduction

Plants are the most precious element of our environment because they are liable to maintain ecological balance and provide us with various foods, fruits and medicines. In Bangladesh, we have many distinct plant species having medicinal and ecological value. Most often environmental scientists or herbal medicine researchers recognize leaves manually by human perception. This will be helpful to them if a computer can recognize the class of the plant simply from images. For this, we feel keen interest to classify leaves available in Bangladesh which have a significant role in herbal medicine and ecology. This will make the recognition of the useful leaves more confident and easier. Leaves recognition will help to specify Ayurveda plants, pollution tolerant plants, endangered species also. All these classifications will help in research in the field of botany, environmental science and medical treatment also.

Leaves recognition previously have done using various machine learning approaches like K-Nearest Neighbor (KNN), Principal Component Analysis (PCA), Probabilistic Neural Network (PNN), clustering (unsupervised learning) etc. All the works leaf feature extraction has proved an efficient method for classification. To classify leaves all required classes of leaves are to use for feature extraction and these extracted features are trained in some classifiers like SVM (support vector machine), KNN, PNN etc. All these methods limit to some issues. Some algorithms work well only for selected species. Again, they require pre-processing steps where human has to give input manually. A convolutional neural network works as a self-learner to extract features to classify leaves.

Recently researchers have proved that newer methods of machine learning are efficient for the classification task. Zhang [1] et al. proposed a seven-layer convolutional neural network to classify 32 classes of leaves. They augmented the dataset applying a various forms of transformation like rotation, translation, scaling, contrast etc. They enlarged the dataset artificially to make the trained model more robust to unseen sample data. They used a network consisting of a convolution layer, a pooling layer and a fully connected layer. The pooling layer caused drop out to prevent the model from overfitting. Using 20 times expanded data he found a classification accuracy of this model up to 90%. Jeon [2] et al. proposed GoogleNet to classify eight different plant species using a leaf image. Inception modules in GoogleNet were used to extract features of the training dataset. The proposed classifier consisted of 22 layers of inception module where multiple convolution was used in parallel on images. To prevent less learning on the training data, he used multiscale training of the images using minimum and maximum sizes. Eight different classes of leaf images were used in their work. Those eight classes were different from each other according to their shapes and features. He improved the classification accuracy up to 90%. Sabu [3] et al. proposed to use two different methods to extract leaves features for classification task. They combined both HOG and SURF features which improved performance than their individual contribution. They trained the features in KNN classifier. After image acquisition, he made a data pre-processing step to prepare data for calculating gradients. Pre-processing steps removed the background area. Morphological operations were done to reduce noise and to sharpen the edge of leaf. In previous works, extra pre-processing steps were needed to get a rotation invariant model. In [1] data was needed to be expanded by augmentation. In case of without augmented data this classification might not be rotation invariant. In [2] multiscaled training dataset was used to improve accuracy which was not possible keeping same dataset. Their limitation was that they had chosen different featured (shape) leaves. In case of similar featured leaves, it is not determined how much the accuracy would still good. In Bangladesh, we are enriched in different useful plants which play an active role in research of herbal medicine and maintaining ecological balance. If an automated system can be used for recognizing these plants, this will be helpful for researchers. To recognize plant species using leaf image we have used supervised learning. Since deep networks are efficient in learning complex features from image data, we are inspired to train Bangladeshi leaf data in deep CNN to make a generalized model to classify leaf.

978-1-7281-3060-6/19/$31.00 ©2019 IEEE

II. METHODOLOGY

We have proposed to follow the transfer learning procedure in order to classify leaves. Using the existing deep network CNN (VGG16, VGG19, Resnet 50, Inception V3, Inception-Resnet v2 and Xception*)* we have trained our leaf dataset. Thus, an existing deep CNN model has initialized its previous weights (as a pre-trained model) and then has started to learn new features of leaves so that the trained model could classify the leaves.

A. Transfer Learning

Transfer learning concerns about solving one problem using source dataset and source task definition. This completed task is stored as knowledge. In machine learning, using the stored knowledge we can solve another problem using our target dataset. Target task depends on the target dataset. This method works in recycling a portion of pre-trained weights. This involves a fully trained network where pre-trained weights are reinitialized and the final classification layer is replaced with custom data training weights according to a new number of classes.

B. Deep Convolutional Neural Networks

Convolutional Neural Network consists of convolution layer, pooling layer and fully connected layer. VGG16 consists of 16 weighted layers. VGG19 differs with it on the number of weighted layers. Both uses SoftMax function as a fully connected layer [5]. Researchers have found adding more layers in deep CNN degrades its final performance. To make a compatible network, Resnet was proposed containing a residual block that represents an identity function [6]. Inception was proposed to reduce the number of parameters to be trained. Smart factorization method is used in Inception. For example, replacing a 5×5 convolution by two 3×3 convolution filters to reduce computational complexity [7]. Xception concerns about modified depth wise separable convolution where depth wise convolution follows pointwise convolution [9]. Inception Resnet v2 is a combination of two important characteristics- residual block and factorizing convolution filters to get better performance. These models differ with each other based on the number of convolution layer, model size, number of parameters which are trained, required training time, number of convolution filters in each layer etc.

VGG16, VGG19, Resnet50, InceptionV3, Inception-ResnetV2 and Xception – these are pre trained model over ImageNet dataset. ImageNet doesn't have any class related to leaf. It contains different classes like fruits, vegetables, animals etc. But we have worked on leaves classes which have much similarities among themselves. Deep CNN requires a large amount of data to learn features. With classification task using a small dataset (per class less than 1000 training samples), it faces a chance to over fit the model or reduce the ability to generalize (recognizing unseen samples) the model. Therefore, we have fine-tuned the models. Fine tuning process involves doing a similar task by a network which has already been trained on a target dataset. All these models were originally trained on ImageNet data to classify 1000 classes. We have tuned the number of parameters of the last layer of the model to replace the output layer of these trained model with our required number of classes. During training in deep convolutional layers, initial layers learn general features. But in deeper layers, it learns patterns which are more specific and complex features are trained efficiently. For this reason, we have kept the initial layers intact making the initial layers freeze. Retraining the later layers with target dataset is done. We have truncated the last layer and replaced this with the relevant new SoftMax layer depending on our dataset.

Algorithm: Transfer Learning

Step 1: Preparing Work Dataset (8 classes of Bangladeshi leaf).

Step 2: Fine tuning the pre trained model to train with leaf dataset and replacing the classification layer (output layer) with new number of classes.

Step 3: Trained model as classifier

III. EXPERIMENT AND ANALYSIS

A. Experimental Configuration

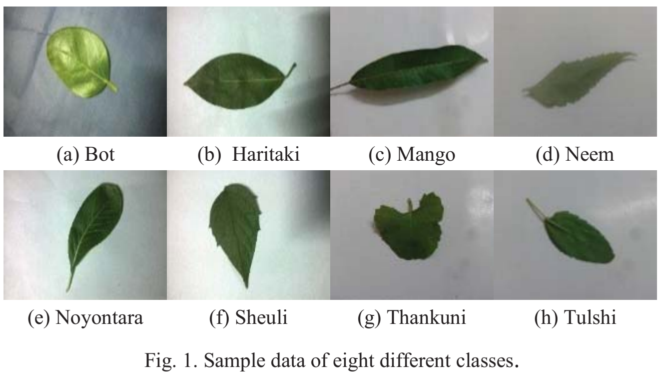
The experiment and analysis are done on the computer having a core i3 processor (having 3.3 GHz speed), 8GB RAM. No GPU memory is used. We have used python (Anaconda). The models are also described in this language.

B. Dataset

Eight different classes of leaves are classified in our work. We have created our dataset based on the Bangladeshi plants. We have prepared our dataset for two categorizes **Ayurvedic plants:** 1. Azadirachta Indica (Neem) 2. Centella Asiatica (Thankuni) 3. Terminalia Chebula (Haritaki) 4. Ocimum Tenuiflorum (Tulshi) 5. Nyctanthes Arbor-tristis (Sheuli) 6. Catharanthus Roseus (Noyontara) **Pollution Tolerant Plants:** 1. Ficus Benghalensis (Bot) 2. Mangifera Indica (Mango). Our dataset contains total of 8000 images of leaves where each class contains 1000 images. We do not train any augmented data rather our dataset contains leaf images from different position and varies with lighting condition. Dataset contains images with different background color also. Bot, Haritaki and Tulshi are oval shape and similarities are found among Neem, Tulshi and Sheuli in their outer small curves. In Fig.1 some sample data of eight different classes are shown.

C. Results and Analysis

We have used deep CNN models for training with our dataset where 80% of the data were used for training and 20% were used as test set for measuring classification accuracy. A portion of data from training set were used for measuring valid-



(a) Bot (b) Haritaki (c) Mango (d) Neem

(e) Noyontara (f) Sheuli (g) Thankuni (h) Tulshi

Fig. 1. Sample data of eight different classes.

ation accuracy. This follows the hold out cross-validation. Training accuracy shows us about the percentage of correctly labelled data at the current training batch. Validation accuracy measures how the percentage of sample data different from training data can be classified correctly during training time. So, validation accuracy is better to measure how generalized the model is. A significant difference between training and validation accuracy shows that the model does not generalize well. During training these six deep CNNs, we have required 8-12 epochs of training to achieve satisfying validation accuracy. We have measured the performance of these trained model using precision and recall calculating from the confusion matrix. Accuracy is a measure that tells how accurate the result is. We have shown the required number of training epoch for six models in Table I. Using the same training dataset, we have trained the models until the validation accuracy starts to degrade. In Fig.2, we have shown that validation accuracy reaches 99% at after 12 epochs of training the VGG16 model. The last three models required less training time than VGG and Resnet. Again, model size is lighter for the last three models in Table I (which use inception module in the network). In the Fig. 3, the confusion matrix of VGG16 on test set is presented which shows that most of the output class labels are positioned diagonally. From Fig.3 we have shown that the number of misclassifications is quite insignificant. In Fig.4 we have found that VGG16, VGG19 and Resnet50 have shown classification accuracy (accuracy is measured on the test data set) over 90% (VGG16 shows maximum). Both precision and recall of VGG16, VGG19 and Resnet50 are above 90%.

All the networks we have experimented, resize the images before training by 224×224. VGG16 as a classifier shows 95.13% accuracy in our dataset. In case of VGG19 it is expected to get better accuracy. But one by one addition of layers forces

TABLE I. NUMBER OF EPOCHS VS VALIDATION ACCURACY.

CNN Architectures	Epoch	Validation Accuracy (%)
VGG16	12	99.96
VGG19	12	98.56
ResNet50	8	92.52
Xception	10	89.93
InceptionResNetV2	12	87.33
InceptionV3	10	85.51

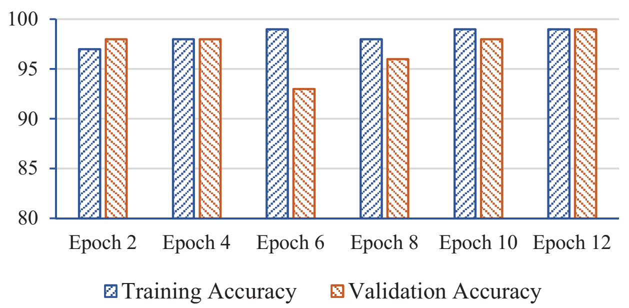


Fig. 2. Validation accuracy vs number of epochs needed during training the model VGG16.

to start degradation of accuracy at 19th layer. So, the accuracy is less than VGG16 here which is 92.75%. When deeper networks start converging, a degradation problem has been exposed: with the network depth increasing, accuracy gets saturated and then degrades rapidly [6]. In Resnet50, the deeper network is replaced and mapped to a shallow network. This change makes the model capable to converge faster than VGG networks. So, training time in Resnet50 is less than VGG but accuracy is decreased a little here which is 89.63%. VGG16 is better comparing with Inception based on the architectural simplicity. But the computational cost is very low in Inception than VGG. With a large data training classification accuracy in Inception outperforms the performance of VGG. Since we have worked with a very small dataset compared to ImageNet dataset, VGG takes more training time and performs better than Inception. Adding a residual layer in Inception, Inception Resnet v2 performs better than Inception on our dataset. The latest model among these six models is Xception, a simplified version of Inception. Xception shows the highest classification accuracy on the ImageNet data. But our small size of dataset causes this model to show less classification accuracy than VGG.

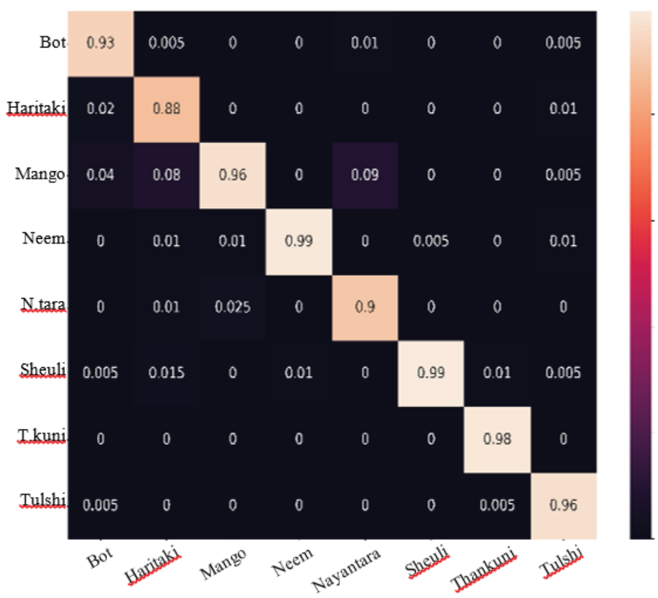


Fig. 3. Confusion matrix for VGG16 as leaf classifier

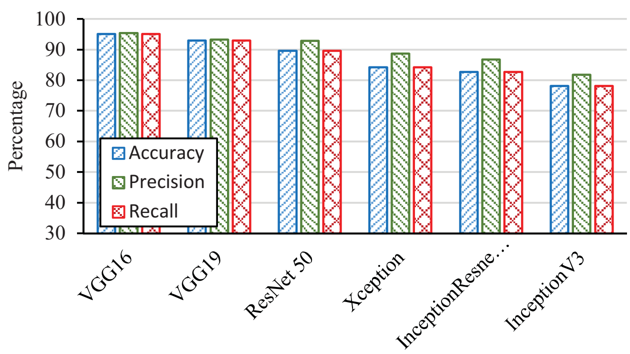


Fig. 4. Classification accuracy, Precision and Recall of six CNN models.

TABLE II. ACCURACY FOR ROTATED TEST IMAGES USING SIX MODELS (IN PERCENTAGE).

	90 Degree	180 Degree	270 Degree
VGG16	91.69	92	91.06
VGG19	91	89.56	90.94
Resnet 50	89.06	89.94	89
Inception V3	67.44	70.13	68.81
InceptionResnetV2	79.06	79.88	79.69
Xception	82.19	83.19	80.81

(a) (b) (c) (d)

Fig.5. Sample misclassification: (a) Haritaki (misclassified as Mango), (b) Mango, (c) Bot (misclassified as Tulshi), (d) Tulshi.

We have tested if the models are rotation invariant or not. Using 90, 180, 270-degree angle of rotation of each test image in VGG16, we have got the classification accuracy respectively 92%, 92%, 91%. The original test set shows the accuracy of 95.13%. So, we have found a trained model that is robust to rotated images. From Table II, we have found that classification accuracy for VGG16, VGG19, Resnet50, Inception v3, InceptionResnet v2 and Xception degrade correspondingly 3%, 2%, 1%, 12%, 3.6% and 2% from original test set accuracy. Hence the trained classifiers are rotation invariant also. Without using any rotated training data, the models show satisfactory performance on the rotated images. We did not use any rotated train image. Two methods can be applied to get good performance on rotated image. Either rotating the feature map of the input images or applying rotation on convolution filters [16]. Since we have not used any of these, classification accuracy on rotated data degrades compared to original test set. But in Inception and InceptionResnet v2, performance degrades more than other models. Because the number of trainable parameters are reduced using inception module [7]. Applying factorization on convolution filters reduces the number of learnable parameters. So, their performance degradation is comparatively high.

Since our dataset contains similarly shaped leaf classes, it causes misclassification of some data. In Fig.5 (a) Haritaki is misclassified and the classifier recognizes it as Mango. The similarity in shape is noticed between (a) & (b). Both Bot and Tulshi are oval shaped. In Fig.5(c) Bot is misclassified predicting it as Tulshi. Our dataset contains 800 training images per class which are very small compared to ImageNet data. Therefore, some data lead to misclassification due to less training data.

IV. CONCLUSION

Leaf classification plays an important role in research in medicinal plants, environmental issues etc. Researchers previously had used feature extraction and then training the classifier. Almost everywhere leaf images required a preprocessing step which had slowed down the classification phase. Adding a new class was so much time-consuming in those methods. In our work, we have used transfer learning in six deep CNN models (VGG16, VGG 19, Resnet 50, Inception v3, Inception Resnet v2, Xception) where our new leaf dataset is trained in a pre-trained model. We have classified 8 different classes of leaves. The classification does not need any pre-processing of images and shows satisfying accuracy. Without training augmented data the model shows robustness on rotated data.

REFERENCES

[1] C. Zhang, P. Zhou, C. Li, L. Liu, “A convolutional neural network for leaves recognition using data augmentation”, IEEE International Conference on Computer and Information Technology, 2015.

[2] W. Jeon and S. Rhee, “Plant leaf recognition using a convolution neural network”, International Journal of Fuzzy Logic and Intelligent Systems Vol. 17, No. 1, 2017.

[3] A. Sabu, Sreekumar K, Nair, “Recognition of ayurvedik medicinal plants from leaves: A computer vision approach”, Fourth International Conference on Image Information Processing (ICIIP), 2017.

[4] J. Chaki, R. Parekh, “Plant leaf recognition using shape based features and neural network classifiers”, (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 2, No. 10, 2011.

[5] K. Simonyan, A. Zisserman, “Very deep convolutional networks for large scale image recognition”, International Conference on Learning Representations (ICLR), 2015.

[6] Kaiming He, Zhang, Ren, Sun, “Deep Residual Learning for Image Recognition”, IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[7] Szegedy, Vanhoucke, Ioffe, Shlens, “Rethinking the inception architecture for computer vision”, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[8] Szegedy, Vanhoucke, Ioffe, “Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning”, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[9] Franc, O. Chollet, “Xception: Deep Learning with Depthwise Separable Convolutions”, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[10] Pushpa, Anand C and Nambiar P, “Ayurvedic Plant Species Recognition using Statistical Parameters on Leaf Images”, International Journal of Applied Engineering Research, Vol. 11, N0. 7, 2016.

[11] Ehsanirad, A. and Kumar, “Leaf recognition for plant classification using GLCM and PCA methods.” Oriental Journal of Computer Science and Technology, Vol. 3, N0. 1, 2015.

[12] Gwo, Wei, “Plant identification through images: using feature extraction of key points of leaf contours”, Applications in Plant Sciences, Vol. 1, No. 11, 2013.

[13] Xiao, Rongxiang Hu, Zhang, and FengWang, “HOG-Based Approach for Leaf Classification”, Springer International Publishing, 2010.

[14] Neto J., Meyer G., Jones D., and Samal A., “Plant species identification using Elliptic Fourier leaf shape analysis”, Computers and Electronics in Agriculture 50, 2006.

[15] Liu W. and He Y., “Fractal features and fuzzy clustering based tree leaf segmentation from images with the complex background”, In Proc. International Conference on Computer, Mechatronics, Control and Electronic Engineering, 2010.

[16] D. Marcos, M. Volpi and D. Tuia, “Learning rotation invariant convolutional filters for texture classification”, 23rd International Conference on Pattern Recognition (ICPR), 2016.