Ayurvedic plant identification using Transfer Learning

FS Sourjah[1#], WMKS Ilmini[2], and WPJ Pemarathne[3]

[123] *Faculty of Computing, General Sir John Kotelawala Defence University, Ratmalana, Sri Lanka*
[#]susnas44@gmail.com

Abstract *– Our healthy ancestors mostly depended on plants around them for their medical reasons. There are hundreds of species of plants around us in our environment which will serve us as home remedies for all most all the diseases and keeps us healthy when taken as daily consumption. Especially in Sri Lanka we have our own set of rare Ayurvedic herbs. But most of us unable to identify these plants due to lack of knowledge. There are existing applications which can identify plants with low prediction accuracies. Also, those applications are based on foreign plant data sets that do not include the valuable herbs and shrubs with medicinal qualities. Hence this research work proposes a mobile application which can identify Ayurvedic plants using Convolution Neural Network models (CNN) with Transfer learning. Three CNN architectures have been applied in this study experimentally to obtain the best accurate model. Ayurvedic plant dataset is trained on CNN from scratch and obtained about 70% of accuracy. Anyhow training a model from scratch is too costly. To overcome this challenge transfer learning has been applied in the Ayurvedic dataset. The pre-trained deep learning models used in this study are Google-Net and Mobile-Net and the accuracies obtained are 80% and 93% respectively. Finally, the Mobile-Net, a small efficient convolution neural network which produces the highest accuracy is used to produce the final prediction model. This model is then used to build a mobile application in the Android platform with TensorFlow-Lite which can identify the Ayurvedic plants using the built-in camera module.*

Keywords: Ayurvedic plants, Mobile-Net, Transfer learning, Convolutional Neural Networks, Mobile application.

I. INTRODUCTION

For over 3000 years form of medication used by more than 75% of the Sri Lankan population is the Ayurveda("Ayurveda in Sri Lanka," n.d.). Its valued as South Asia's ancient health care system based on herbs and diet. Sri Lanka is naturally rich country about 3500 species of plants and about one quarter of that is endemic to the country. Among them there exist about 1344 species of plants with rich ayurvedic properties where some of them are endemic to Sri Lanka. But Most of us, the public fail to identify these plants while the knowledge remains unshared with the ayurvedic doctors and botanists. For about few decades with advancement in the western medicine people used to only rely on them forgetting the traditional Ayurveda. But the health-conscious today is searching for an effective alternative to the high cost and side effects that result from the use of modern medicine. Which is why the Ayurveda again has gained its prominence. There for people should gain the knowledge about the plants and herbs around them in the environment so that they can use them in their day -to day life. There exist many industrial herbal companies which rely on these natural herbs to produce their products. It's very important for them to identify correctly their raw materials, these Ayurvedic herbs. Mistakes in identifying these plants could lead to life threatening issues. An automatic plant identification system will be the best solution rather than hiring specialists to detect these plants and waste hours of time.

A deep learning approach can be used to build a system which can predict these plants. Training a model from scratch is very difficult since it needs large amount of data and many hours of time. The amount of data that can be collected easily depends on the type of the problem. Specially it's very difficult to find large amount of data in this type of problem since it's about the rare type of natural herbs and plants. There for this paper suggests of using the Transfer learning (Krishnaswamy Rangarajan et al., 2019) which is the technique of training pre-trained models on new data to acquire high accuracy. Mobile Net (Howard et al., 2017) which is a small, efficient and mobile compatible convolutional neural network will be used to train the new dataset to create a machine learning model. Mobile Net is a pretrained network on the Image net, the world's largest dataset with thousands of different kinds of images. The newly created model is converted to a mobile compatible model to create the android application to predict Ayurvedic plants.

II. LITERATURE REVIEW

The research work (Kumar et al., 2017) classifies the Ayurvedic medicinal plants using image processing techniques based on the feature combination of the leaves. The dataset used by this research work consist of scanned images of 20 leaves from 40 different plant species. Both front and back of the leaves were scanned to capture the unique feature combination. This work presents their model to identify the leaves as two separate phases which are training and testing phases. During the training phase first the images are pre-processed to give all the images the same resolution. Next the images are converted into binary images. The feature extraction process then extracts the features from both colour and binary images and stores the value in a feature table. These feature values are trained using a classifier. This work states that this classification of medicinal plants has about 99% accuracy with an ability to identify even dried leaves. But this research work doesn't much explain about the classifier used to train the model and doesn't focus on how they achieve such a high accuracy rate.

The research study (Hedjazi et al., 2017) uses transfer learning which is a deep learning technology of using a pretrained model to train new data to eliminate the difficulty faced of having small datasets. This study uses a pretrained Convolutional neural network model on Image Net to train a small dataset of plants. The data set consist of leaf images of plant taken both in natural background and a uniform background. This research work adheres to an experimental procedure to achieve the final better accuracy. The work first achieves an accuracy of about 83% using traditional machine learning. Then this work uses different versions of Alex Net which is a pretrained model. The major differences between these versions of Alex net is the fine-tunning. Fine tuning can improve the accuracy of the Alexa Net on the same dataset. Finally, they have achieved an accuracy of about 96% using these fine-tuned Alex Net. This work focuses on explaining how transfer learning can out performs the traditional ML approaches to achieve better accuracy with smaller dataset.

The deep learning approach using convolution neural network to identify flower species with better accuracy is proposed by the research work (Gogul and Kumar, 2017).This work uses CNN combined with transfer learning approach to achieve the accuracies of 73%,93% and 96% using Over Feat, Google Net and Inception architectures, respectively. The overall process adhered by this research work can be divided into three parts. First the features of the flowers are extracted using any one of three CNN networks. Next the machine learning classifiers like Logistic Regression, Decision Trees, Random Forests and Stochastic Gradient Boosting are used to train the network. The last part is the testing part to evaluate the accuracy. The flower dataset set consists of about 1680 images. One of the disadvantages of this work is that they have used many images even though their approach for this problem is transfer learning.

The research work (Krishnaswamy Rangarajan et al., 2019) presents a methodology to identify the grape crop diseases using deep learning technologies. The classification is done using convolution neural networks combined with transfer learning approach. This work uses Alex Net a pretrained architecture to train about 4063 images of grape crop to classify among three grape crop diseases. The model achieves about 97% classification accuracy. Alex Net consist of 5 CNN layers. The weights and learning rates of previous layers in the architecture are not change except for the last layer.

The paper (Sun et al., 2017) focuses on Accuracy of recognition of about 91.78% using a dataset and using deep learning technology. The paper also talks about all kinds of solutions used for the classification from time to time like BP neural network, CNN (convolution neural network) and so on. But those techniques had some challenges which need to be solved. To overcome the challenges the paper proposes data set containing 10,000 images and a 26-layer deep learning model. Unlike the traditional methods this new proposed layer improves accuracy and solves the challenge of degradation present in earlier models. Good accuracy and nice convergence behaviors can be obtained from Deep residual networks with residual units on several large-scale image recognition tasks. Data set is trained well to have very high accuracy rates where 80% of the dataset is selected as training set while the remaining as test set. But the paper doesn't choose a validation set which is very important. Overfitting is one challenge that can mislead us. Choosing a validation set to evaluate the results from training set. Then use the test set to double check the evaluation after the model has passed the validation set is very important. This can be a problem of this proposed idea.

The research study (Taghavi Namin et al., 2018) propose CNN-LSTM framework for plant classification of various genotypes. Here a deep CNN is used additionally LSTMs (Long-Short Term Memories) to study growth of plants. A dataset with time series image sequence are used to train the model. The use of LSTMs further improves the performance of the system. This proposed framework is a good solution to the plant classification.

The research study (Zhao et al., 2014) classify the plants using a growing convolution neural network which can learn generic features of plants, directly acting on the two-dimensional image pixels without changing the topology of the input images. This network has a simple architecture where each layer is added with new neurones, and the weight is modified so that a very low loss is gained until a better training accuracy is gained. This set of networks uses weight-sharing technologies which reduces the number of free parameters and can be used to detect the representation in different angles. This paper proposes a growth algorithm to construct to construct the CNN rather a traditional CNN which is of high cost and complex. The growing CNN can grow up by itself. This work first classifies plant leaf images using the traditional CNNs.Next Uses CNN with growing structure to classify plant leaf images. The third method used is a CNN with growing structure and progressive sample learning method to classify leaf images. finally using other state-of-the-art methods. This paper focuses on using an alternative for traditional CNN which is cost effective but not about the accuracy of prediction.

The research work(Gwo and Wei, 2013)propose a feature extraction method for leaf contours' study focuses on 92.7% of accuracy. The study uses a leaf recognition framework which is divided into leaf modelling and leaf recognition. A model for each leaf in the database is created using the extracted feature. The leaf recognition is about detection of a leaf by feature points and feature extraction. The specific feature of this paper is the extracted feature are invariant to scale and rotation. Multiple leaf templates are suggested for creating the species of leaf model.

The research paper (Lee et al., 2015) is a study on plant identification using convolutional neural networks (CNN) by learning the features of the plant leaves. The study focuses on tow methodologies namely the bottom-up and top-down methods of deep learning. Very important part of the study is using a deconvolutional networks (DN) to visualize the learned features. First a CNN model is proposed to learn the features of the plants and later the feature representation is learnt by DN. The study gives of about 99.6% of accuracy in prediction. But the study fails to learn about how to gain efficiency of the system reducing the cost and computation.

III. DEEP LEARNING MODULES

The performance of the following three deep neural networks are evaluate for the Ayurvedic medicinal plant identification to achieve the better accuracy.

A. Traditional CNN from scratch.

The CNN model (Allibhai, 2018) used here in this study takes the images of the plant leaves raw pixel data as the input instead of preprocessing the data to extract features. Each module in a CNN consists of three operations("ML Practicum," n.d.). First a convolution extracting tiles of the input feature map, and applies filters to them to compute new features, producing an output feature map. Next a Rectified Linear Unit (ReLU) transformation is applied by CNN to the convolved feature. This introduces nonlinearity into the model. The last is the pooling step to reduce the number of dimensions of the feature map while still preserving important features. At the end of CNN are the fully connected layers. These layers use the features extracted by the CNN to perform the classification. Figure 1 shows the architecture of the CNN

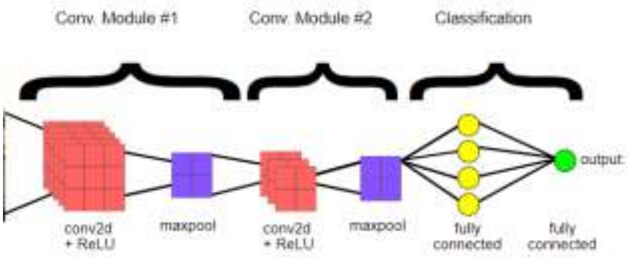

Figure 1. CNN Architecture

B. Google Net, a CNN pretrained on ImageNet ILSVRC dataset.

This is a transfer learning methodology where the last fully connected layer of the pretrained CNN is removed completely and treat the previous layers as a feature extractor for the new Ayurvedic plant dataset *(Chu, 2017)*. In a convolutional operation every output channel is connected to every input channel. But the google net is developed based on the idea that some of these connections are unnecessary because of the correlations between them. Therefore, here in this Google Net (Prabu, 2018) all the output channels are not connected to all the input channels rather there are techniques to prune out such connections which would result in a sparse weight/connection. This network is very efficient with small number of neurons with convolutions of different sizes to capture details at varied scales (5X5, 3X3, 1X1). Google net*(Szegedy et al., 2015)* achieves 93.3% accuracy on image net and much faster.

C. Mobile Net

Since this study proposes a mobile application to identify the Ayurvedic medicinal plants it's very important to choose a ML architecture which is compatible with mobile platform. Therefore, this work uses the Mobile Net a lightweight efficient CNN architecture that supports mobile vision applications. It uses depth wise separable convolutions which basically means it performs a single convolution on each color channel rather than combining all three and flattening it. This network consists of 30 CNN layer with 2 strides, depth wise layer,pointwise layer that doubles the number of channels,depthwise layer with stride 2 and point wise layer that doubles the number of channels (Culfaz, 2018).Training on Mobile Net is very efficient with smaller datasets and training time is very low compared to any other networks(Tsang, 2018),(Xu, 2019) . Figure 2 shows how Mobile Nets achieve higher accuracy within lesser time.

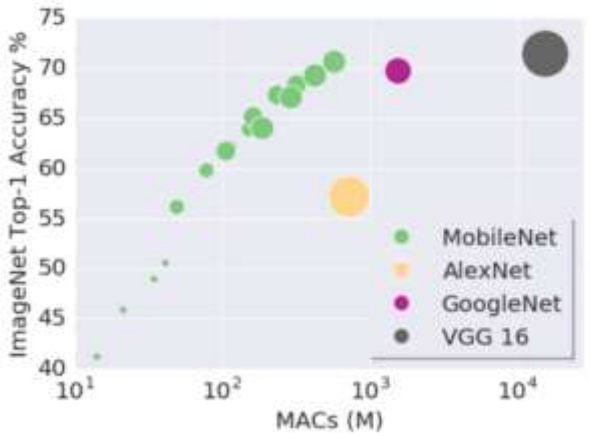

Figure 2. Comaparision of Mobile Net with other network
Source: *(Culfaz, 2018).*

IV. METHODOLOGY

A. Dataset

Ayurvedic medicinal plant dataset consist of about 1500 images belonging to 10 different species of plants. The pictures of the plants are taken in the natural background. All the pictures are taken originally, visiting herbariums and home gardens. When selecting these 10 species for this research work the availability of these plants in the home garden is taken in to consideration. The traditional home gardens in Sri Lanka have lots of different species of medicinal plants grown even without the knowledge of people. Most people fail to identify these plants in their garden due to lack of knowledge on plants.

B. Modules for the proposed system

The Ayurvedic medicinal plant dataset is trained using a CNN from scratch, Google Net and Mobile Networks to achieve a better accuracy. The data set was divided into three as training set, validation set, and testing set in the ratio of 70:15:15.

When training using the CNN from scratch one of the main concerns is the overfitting. That is the model is tuned so much to the training specifics and unable to produce results for the new data. Two techniques used to avoid the overfitting are Data augmentation and Dropout regularization. Data augmentation is a technique where to increase the dataset by performing various transformations to the images like rotating and flipping. These techniques are useful since the Ayurvedic plant dataset is relatively small. Producing feature cross results in the network can increase the complexity of the network and require a huge amount of RAM. Zeroing out these features can help to reduce the complexity of the network. That's dropping the weights of unnecessary features to 0 help save the RAM. Dropout regularization removes a random selection of a fixed number of the units in a network layer for a single gradient step to prevent overfitting. The feature extraction in each convolution unit is in the Figure 3.

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	(None, 150, 150, 3)	0
conv2d (Conv2D)	(None, 148, 148, 16)	448
max_pooling2d (MaxPooling2D)	(None, 74, 74, 16)	0
conv2d_1 (Conv2D)	(None, 72, 72, 32)	4640
max_pooling2d_1 (MaxPooling2	(None, 36, 36, 32)	0
conv2d_2 (Conv2D)	(None, 34, 34, 64)	18496
max_pooling2d_2 (MaxPooling2	(None, 17, 17, 64)	0
flatten (Flatten)	(None, 18496)	0
dense (Dense)	(None, 512)	9470464
dense_1 (Dense)	(None, 1)	513

Total params: 9,494,561
Trainable params: 9,494,561
Non-trainable params: 0

Figure 3. Feature extraction in CNN

The "output shape" column shows how the size of the feature map evolves in each successive layer. The convolution layers reduce the size of the feature maps by a bit due to padding, and each pooling layer halves the feature map. The model produced using this approach only produced an accuracy of about 70%.

Training the CNN model needed a large amount of data, which was produced using data augmentation and the process of training is time consuming. To overcome this challenge and to achieve better accuracy than before the technique of transfer learning was used. The plant dataset was trained using Google Net network. The early layers of the Image net model are frozen to act as feature extractor to extract the shape, colour and textures of the leaves.

Final layer is trained for the new dataset. To increase the accuracy number of CNN units in the final layer was increased. The system could achieve an accuracy of about 83% using this network.

C. Final Module for the proposed system

The non-functional requirements should be considered apart from the functional requirements when using the machine learning models for the implementation of a software. Considering the efficiency, size and compatibility of Mobile Nets with Android applications the plant dataset is trained again using Mobile Net. No data augmentation techniques are used on the dataset. Original dataset is used since the Mobile Net can even produce better accuracy for relatively smaller datasets. Mobile Net is pre-trained for Image Net dataset and Image Net doesn't consist any of these Ayurvedic leaves. The earlier layers of this net are trained to classify thousands of classes. This information is used as an input to the last classification layer to classify these plant species. Creating bottlenecks for each Ayurvedic plant species is very important to save the time when the training script is rerun for more training. Since the lower layers of the net are not modified when the images are run multiple time through these lower layers their outputs be cached and reused using the bottleneck values. When there's a need of retraining the dataset, these bottle neck values can be reused without waiting to produce these outputs again. This makes the model more efficient. After creating the bottleneck values the training of the final layer starts. The layer is trained for 500 epochs. At each step bottleneck values of 10 random images are found and fed into the final layer producing predictions. These produced predictions are then compared with labels of the plant and the results are compared to change the final layer weights of the Mobile Net using the backpropagation algorithm. The following graphs in Figure 6 show how the train, validation accuracies and cross entropy changes with function progress.

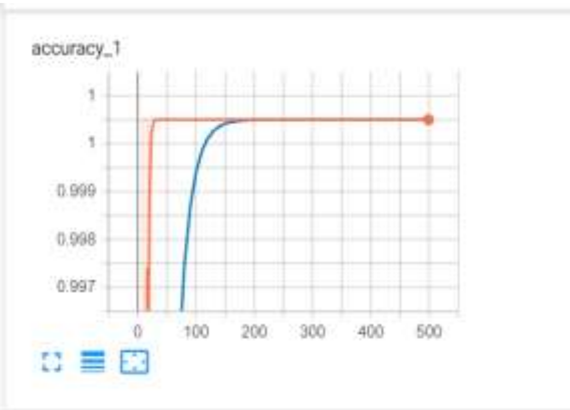

Figure 4. Accuracies and Cross entrophies

When training the network adjusting the learning rates to various values is an important process to achieve better final validation accuracy. Learning rates are changed from highest value like 0.01 to a lower value like 0. 005.Having lager learning rates the network can train faster. Having lower learning rates could consume more time but will produce higher accuracies. The final produced module can classify the leaves as in the Figure 7 below.

```
Evaluation time (1-image): 1.031s

kapparavalliya (score=1.00000)
thippili (score=0.00000)
aththora (score=0.00000)
thebu (score=0.00000)
adathoda (score=0.00000)
```

Figure 5. Predictions

V. POPOSED SYSTEM

A. System Overview

The TensorFlow("TensorFlow," n.d.) module produced from training the Mobile Net is then converted to TensorFlow lite model to create an android application which can classify these Ayurvedic plants. Figure 8 shows the system overview of the developed system.

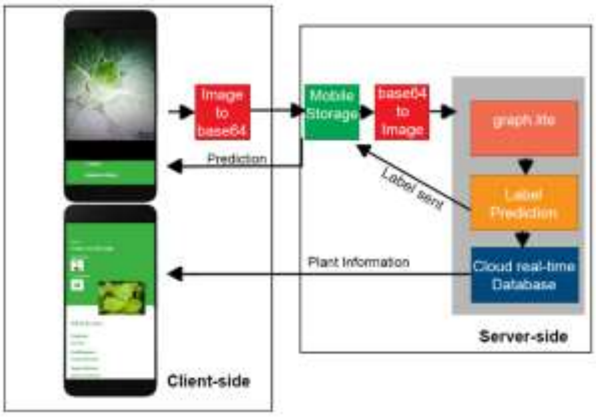

Figure 6. System Overview

This app uses pre-compiled TFLite Android Archive (AAR)("TensorFlow Lite guide | TensorFlow Lite," n.d.). The maven uses this AAR from the TensorFlow binary maven repository. TFlite model is bundled with the application in the assets folder. The image taken using the app is first converted in to float. The interpreter initialized to load the bundled model accepting float array values. Then the model runs passing the input buffers and output arrays as arguments. The interpreter sets the values in the output array to the labels of the corresponding images to be written in the app interface. The app then displays the details of the corresponding plant.

B. Software Requirements

Python 3.6 ("3.6.8 Documentation," n.d.) and TensorFlow("TensorFlow," n.d.) library are used to build the CNN models and retraining scripts. The TFlite converter is used to convert the tensor model to TensorFlow lite model. Other libraries like Pillow, NumPy, SciPy, matplotlib ("Numpy and Scipy Documentation — Numpy and Scipy documentation," n.d.) libraries are used for retraining purposes of the Mobile Net. The android application is created using java where its connected to the google Firebase cloud storage.

VI.RESULTS AND DISCUSSION

The overall plant identification problem presented by this work adheres to an experimental procedure. First the Ayurvedic plant dataset is trained using a CNN from scratch, secondly the dataset is trained using Google Net Inception V3 pretrained model and finally using the Mobile Net_0.50_224.The final accuracies are then compared to use the best model in the android application. The comparisons of the 3 models are shown in the table 2 below.

Table 1. Comparison among the models

models	Accuracy	Time taken to train
CNN (scratch)	70%	high
Google Net	83%	Comparatively low
Mobile Net	93%	Very low

When selecting the most suitable model among the above three architecture not only the better accuracy is considered. Other non-functional requirements like efficiency, size, compatibility and power consumption are also taken into consideration.

VII. CONCLUSION AND FUTURE WORK

The implemented Ayurvedic plant identification mobile application will be very useful in identifying the Ayurvedic plants around us in our home garden. Also, this research work will greatly influence the people the use of Ayurvrdic plants in their day to day lives. There are hundreds of species of Ayurvedic plants on this earth. The same methodology can be easily adapted for the scalability of this app to identify all the Ayurvedic plants in the future. The full implementation of the application with a map where the locations of the availability of these plants will be expected as the future work.

ACKNOWLEDGMENT

Authors would like to acknowledge everyone who has supported in conducting this research and all the reviewers, whose remarks helped improve this article substantially.

REFERENCES

3.6.8 Documentation [WWW Document], n.d. URL https://docs.python.org/3.6/ (accessed 5.28.19).

Allibhai, E., 2018. Building a Convolutional Neural Network (CNN) in Keras [WWW Document]. Data Sci. URL https://towardsdatascience.com/building-a-convolutional-neural-network-cnn-in-keras-329fbbadc5f5 (accessed 5.28.19).

Ayurveda in Sri Lanka [WWW Document], n.d. URL https://lanka.com/about/interests/ayurveda/ (accessed 5.28.19).

Chu, G., 2017. How to use transfer learning and fine-tuning in Keras and Tensorflow to build an image recognition…. Medium. URL https://medium.com/deeplearningsandbox/how-to-use-transfer-learning-and-fine-tuning-in-keras-and-tensorflow-to-build-an-image-recognition-94b0b02444f2 (accessed 5.21.19).

Culfaz, F., 2018. Transfer Learning using Mobilenet and Keras [WWW Document]. Data Sci. URL https://towardsdatascience.com/transfer-learning-using-mobilenet-and-keras-c75daf7ff299 (accessed 5.21.19).

Gogul, I., Kumar, V.S., 2017. Flower species recognition system using convolution neural networks and transfer learning, in: 2017 Fourth International Conference on Signal Processing, Communication and Networking (ICSCN). Presented at the 2017 Fourth International Conference on Signal Processing, Communication and Networking (ICSCN), pp. 1–6. https://doi.org/10.1109/ICSCN.2017.8085675

Gwo, C.-Y., Wei, C.-H., 2013. Plant identification through images: Using feature extraction of key points on leaf contours1. Appl. Plant Sci. 1. https://doi.org/10.3732/apps.1200005

Hedjazi, M.A., Kourbane, I., Genc, Y., 2017. On identifying leaves: A comparison of CNN with classical ML methods, in: 2017 25th Signal Processing and Communications Applications Conference (SIU). Presented at the 2017 25th Signal Processing and Communications Applications Conference (SIU), pp. 1–4. https://doi.org/10.1109/SIU.2017.7960257

Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H., 2017. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. ArXiv170404861 Cs.

Krishnaswamy Rangarajan, A., Raja, P., Aniirudh, R., V. Mukesh, K., Ashiwin, R., Vikas, G., 2019. Grape Crop Disease Classification Using Transfer Learning Approach, in: Lecture Notes in

Computational Vision and Biomechanics. pp. 1623–1633. https://doi.org/10.1007/978-3-030-00665-5_150

Kumar, P.M., Surya, C.M., Gopi, V.P., 2017. Identification of ayurvedic medicinal plants by image processing of leaf samples, in: 2017 Third International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN). Presented at the 2017 Third International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), pp. 231–238. https://doi.org/10.1109/ICRCICN.2017.8234512

Lee, S.H., Chan, C.S., Wilkin, P., Remagnino, P., 2015. Deep-Plant: Plant Identification with convolutional neural networks. ArXiv150608425 Cs.

ML Practicum: Image Classification | Machine Learning Practica [WWW Document], n.d. . Google Dev. URL https://developers.google.com/machine-learning/practica/image-classification/convolutional-neural-networks (accessed 5.21.19).

Numpy and Scipy Documentation — Numpy and Scipy documentation [WWW Document], n.d. URL https://docs.scipy.org/doc/ (accessed 5.28.19).

Sun, Y., Liu, Y., Wang, G., Zhang, H., 2017. Deep Learning for Plant Identification in Natural Environment. Comput. Intell. Neurosci. 2017. https://doi.org/10.1155/2017/7361042

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., 2015. Going Deeper with Convolutions, in: Computer Vision and Pattern Recognition (CVPR).

Taghavi Namin, S., Esmaeilzadeh, M., Najafi, M., Brown, T.B., Borevitz, J.O., 2018. Deep phenotyping: deep learning for temporal phenotype/genotype classification. Plant Methods 14. https://doi.org/10.1186/s13007-018-0333-4

TensorFlow Lite guide | TensorFlow Lite [WWW Document], n.d. . TensorFlow. URL https://www.tensorflow.org/lite/guide (accessed 5.28.19).

TensorFlow [WWW Document], n.d. . TensorFlow. URL https://www.tensorflow.org/ (accessed 5.28.19).

Xu, Y., 2019. MobileNet: Paper Review and Model Architecture. Yinghan Xu. URL https://medium.com/@rockyxu399/mobilenet-paper-review-and-model-architecture-7963c22ea528 (accessed 5.25.19).

Zhao, Z.-Q., Xie, B.-J., Cheung, Y., Wu, X., 2014. Plant leaf identification via a growing convolution neural network with progressive sample learning, in: Asian Conference on Computer Vision. Springer, pp. 348–361.