



INTERNATIONAL RESEARCH JOURNAL ON ADVANCED SCIENCE HUB

e-ISSN : 2582 - 4376
Open Access

RSP SCIENCE HUB

(The Hub of Research Ideas)

Available online at www.rspsciencehub.com

Special Issue of First International Conference on Information Technology, Computing & Applications (ICITCA 2021)

Medicinal Plant Identification Using Deep Learning

R.Geerthana¹, P.Nandhini², R.Suriyakala³

^{1,2,3} Department of Computer Science & Engineering, Velammal College of Engineering and Technology
Tamilnadu, India.

geerthanasushmi@gmail.com¹, nandhini.pandiarajan2000@gmail.com², suriya08be@gmail.com³

Abstract

In this paper, our main aim is to create a Medicinal plant identification system using Deep Learning concept. This system will classify the medicinal plant species with high accuracy. Identification and classification of medicinal plants are essential for better treatment. In this system we are going to use five different Indian medicinal plant species namely Pungai, Jamun (Naval), Jatropha curcas, kuppaimeni and Basil. We utilize dataset contains 58,280 images, includes approximately 10,000 images for each species. We use leaf texture, shape, and color, physiological or morphological as the features set of the data. The data are collected by us. We use CNN architecture to train our data and develop the system with high accuracy. Several model architectures were trained, with the best performance reaching a 96.67% success rate in identifying the corresponding medicinal plant. The significantly high success rate makes the model a very useful advisory or early warning tool.

Keywords: Deep Learning, Neural Networks, Convolutional Neural Network, Regression.

1. Introduction

A medicinal plant is delineated as a plant that is collected from the wild or planted plant for its medicinal value. Plants have been utilized in curing human diseases for thousands of centuries and are the source of a significant percentage of medicines. Medicinal plants have a prolonged history of consumption with hundreds of classes. In recent years, some computational approaches have been introduced, particularly in image processing domain, for plant classification. In this regard, Neural Networks represent novel techniques for image processing, with large potentials. The most commonly used Neural Networks for image processing is Convolutional Neural Network (CNN).[1-5].

1.1 Motivation

Medicinal plants have been utilized in curing human diseases for thousands of centuries and are the source of a significant percentage of medicines. Medicinal plant species classification is

critical for medicine production and conservation. Local peoples are not enough knowledgeable of their urban medicinal plants and their usages. Therefore, classifying the medicinal plant image using Convolutional Neural Network by high accuracy image classification model could be useful to identify different types of species.

2. Existing System

There is no particular research has been done to detect medicinal plants of Canada using image classification technique with the best of our knowledge. However, there are different proposed techniques to classify objects or flowers, and some of them employed deep learning approaches. In most researches, classification challenges are deal with using CNN- based algorithms. The Convolutional Neural Networks (CNN) shows considerable accomplishment in different research and real-world projects. One of the earliest applications of CNN was handwritten recognition [2].By developing of CNN technique, innovative

models, which include new layers, are presented. In ImageNet challenges, CNN has been utilized mostly with different datasets mixtures. Some researchers compared human detection abilities and a trained network on image recognition. The output of these comparisons depicts that humans can detect an image with accuracy about 73.1% while the trained network shows up 64% accuracy. Also, after employing Convolutional Neural Networks to a similar dataset, it revealed the 74.9% accuracy, so it depicts better results than human recognition.

3. Data

Numerous medicinal plant species are available. In this system we are going to use five different Indian medicinal plant species namely Pungai, Jamun (Naval), Jatropha curcas, kuppaimeni and Basil. we utilize dataset contains 58,280 images, includes approximately 10,000 images for each species. We use leaf texture, shape, color, physiological or morphological as the features set of the data. The data are collected by ourselves. We use CNN architecture to train our data and develop the system with high accuracy. Several model architectures were trained, with the best performance reaching a 96.67% success rate in identifying the corresponding medicinal plant. The significantly high success rate makes the model a very useful advisory or early warning tool. Basil most popularly known as Tulasi has been used for thousands of years in ayurvedha for its diverse healing properties. The Leaves strengthen the stomach and help in respiratory diseases. It is also used to cure diabetes, stress, Kidney stones. Traditionally Kuppaimeni is used to get rid of unwanted hair growth, for treating skin problems like pimples, psoriasis to get rid of cough and cold, to treat intestinal worms and also for treating piles. Medicinal values Jatropha provides is to treat skin, cancer, digestive, respiratory and infectious diseases. Jamun, the highly nutritious, refreshing and succulent fruit flooding the summer markets has innumerable health benefits. Commonly known as Java plum or Indian blackberry in English. Ayurveda strongly recommends this berry for treating various conditions related to heart, arthritis, asthma, stomach pain, bowel spasm, flatulence and dysentery. The diuretic effects of jamun flushes toxins out of the kidneys, while the high fibre content aids in digestion and prevents

nausea and vomiting. Pungai leaves used as a digestive and laxative and to treat inflammation and wounds. Leaf juice aids in treatment of Leprosy, gonorrhoea, diarrhoea, cough and cold. In this study we follow some steps for data collection.

i) Articulate the problem; knowing what one wants to predict helps in deciding the data valuable to collect. Data Exploration in the categories of Classification, Clustering, Regression, and Ranking helps with the decision.

ii) Establish Data Collection Mechanism; Process of collecting the Data which can be Automated or Manual based on the requirement.

iii) Format Data; File format of the images stored need to be same for maintaining the consistency.

iv) Reduce Size; Data need to be collected based on the target needs to be achieved which is critical for our Dataset.

v) Complete Data Cleansing; Data with missing, erroneous or fewer representative values is removed to make prediction more accurate.

4. Background study & Methodology

Background Study

Convolutional Neural Network (CNN): A Convolutional Neural Network, CNN, is a deep learning architecture. Image classification is one of the problems that a CNN could do and is a trained network that can classify images into one of a thousand pre-determined categories. One can employ a CNN to do image processing including image detection, segmentation, and classification. The main advantage of CNN compared to a typical NN is that it automatically detects the significant features without any supervision. A CNN consists of various layers that transform an input into the output. The complexity of the learned features increases in every hidden layer. For example, detection simple features are learned in the first hidden layer, like edges, and the detection of more complex shapes in the last one. A CNN model is composed of two main components: the feature extraction part and the classification part. In order to understand CNN architecture, we introduce several concepts. Compared to a typical neural network in which each input layer's neuron is linked to the hidden layer's neurons. In a CNN we have Local Receptive Fields, a small number of input layer's neurons which are connected to the hidden layer's neurons. The local receptive field use convolution to translate an image into a feature

map. Convolution can perform the mathematical convolution operation by moving a filter across the image. At every region, an element-wise matrix multiplication and summation of the result are done. Non-linearity features make powerful any type of neural network. A neural network can achieve this using an activation function by passing the weighted sum of its inputs to the next layer. CNN use the same function and applies the transformation to the output of each neuron by passing the result of the convolution operation

through an activation function. ReLU or Rectified Linear Unit is a popular activation function that maps the output of a neuron to the highest value in the next layer. A pooling step can use for the dimensionality reduction of the features map by compressing the output of small number of neurons into a one output. We call this a kind of down-sampling the feature map which keeps the important features on CNN automatically. Meanwhile, this can leads to reduction of the number of parameters to learn the model.[6-10].

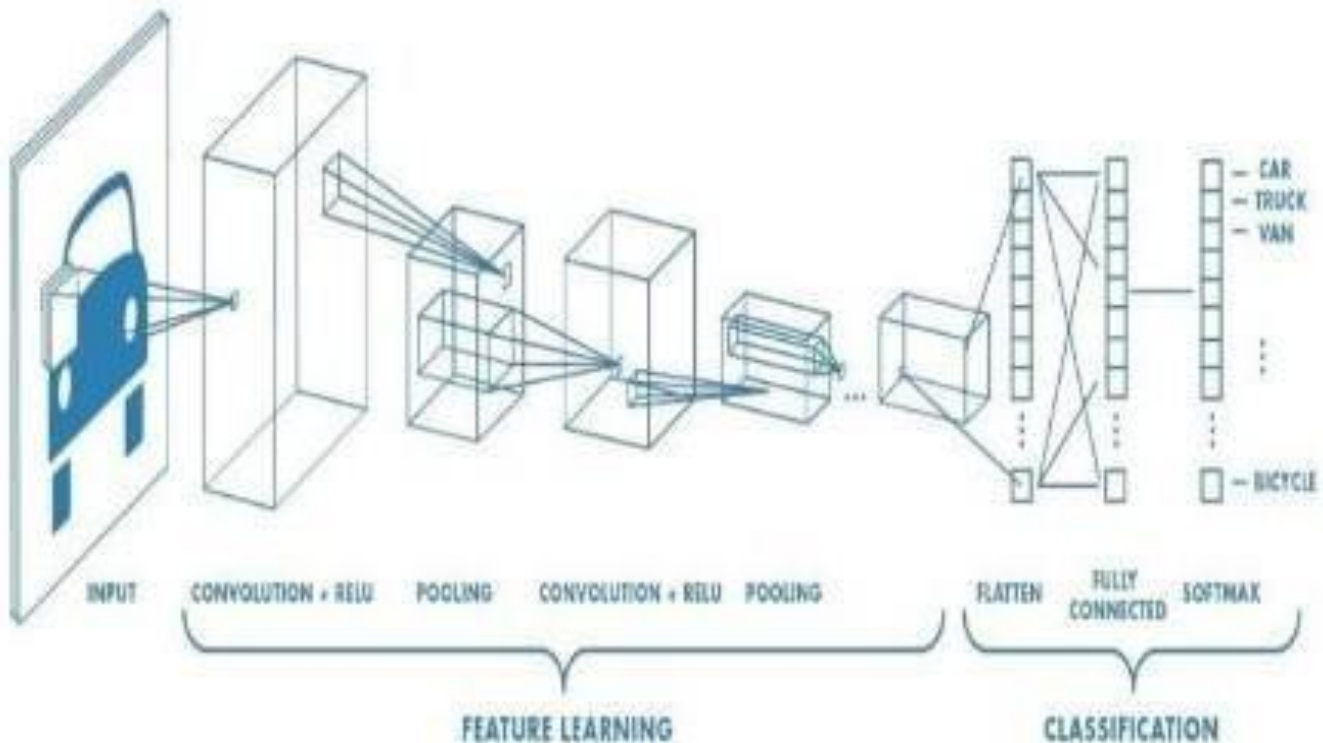


Fig. 1. The Overview of a CNN and its main components

5. Architecture

The architectural design consists of five different phases in it. Where each phase is responsible for its own task, starting from the setting up of image directory to the final output of the model. Flow from directory - This is the first phase of our model. Here the directory of the images resides are selected and introduced into the model for fetching and further proceedings. After they are selected, they are subjected to transformations, scaling. Model Setup- This is the second phase, here the architecture for our model is 23 being setup. We are using an CNN architecture, so we need to define all the layers and parameters. The building up of the entire architecture is taken up in this phase. Loss function and Optimizers - In this phase

the loss function which is going to be used up in our model is begin setup and also the optimizers are setup. This phase is the most important phase , because based up on the type of loss function and optimizer only the model's efficiency is found. Callbacks - Callbacks are nothing but defining, "what has to be done if the model is learned a very good accuracy values", and "what has to be done if our model has to improves it learning rate if it has to gain accuracy". Here depending upon the number of epochs and steps defined inside the epochs. The callback will be executed. Saving the Model - This is the final phase where our model is given an file name for saving the model. So that during testing we can run by using this saved file.

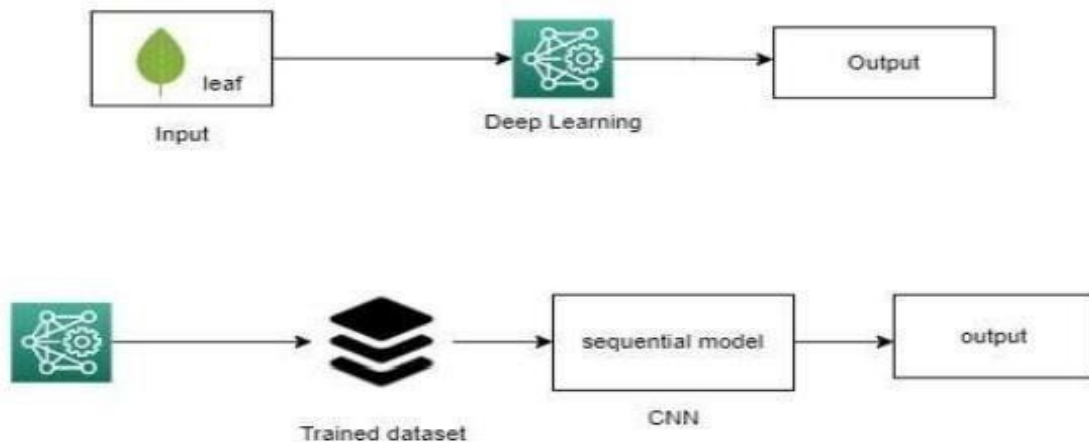


Fig.2.Workflow Diagram

6. Implementation

In this section, we explain model construction, compilation, and validation. To construct the model, we use InceptionV3model as a pre-trained model/base model. The model is available in keras application library. The model construction includes the following steps:

- ✓ First, we initialize InceptionV3model as our base model. The parameters trained on Image Netdataset.
- ✓ Then, we add a Global Average Pooling (GAP)layer to the model to reduce the spatial dimensions of a three dimensional tensors, which eventually leads to minimize overfitting.
- ✓ It reduces $h * w * d$ dimensions to $1*1 * d$ dimensions by taking the average of h and w values. Then, we add a fully connected layer, FC, of 1024 hidden nodes.
- ✓ Finally, the FC layer is connected to our output layer to predict output.

The FC layer mainly takes the output from GAP as input and provides output to our final output layer to predict the best label for each image. We use ‘Softmax’ as the activation function, because the numbers of output classes are more than two. We use 65 layers in the model and make 249 layers non- trainable. The summary of the constructed model is provided in II. In this section, we explain data distribution, data augmentation, and data rescaling and resizing. First of all we divide dataset into three subsets including train, test, and validation set. The initial dataset was included 859 images. We separate 80% of it for training, 10% for test, and10% for validation set. The size of each class is shown in Table 1. Also, the

distribution of each subset is displayed in Fig. As we can see, the proportion of each class equals the three subset.

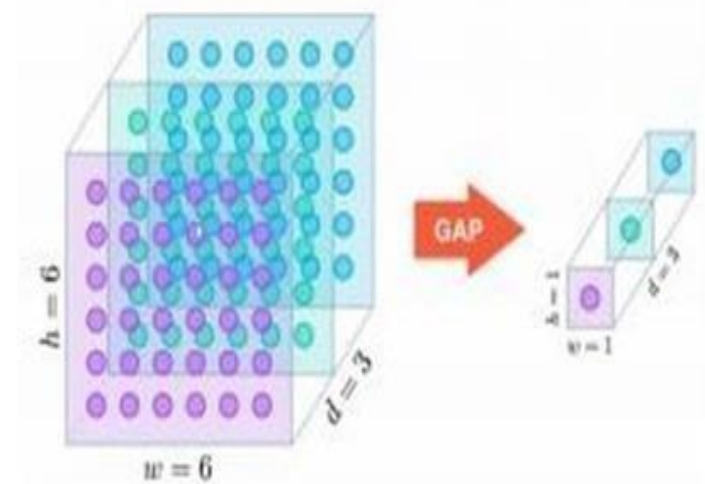


Fig.3.Reduce of the spatial dimensions of a three dimensional tensors using Global Average Pooling layer(GAP).

Table.1.The size of the samples in Train, Test and Validation

	Jatroph a	Kuppa imeni	Naval	Pungai	Basil
Train set	11323	10589	14165	10624	11579
Test set	50	50	50	50	50
Valid set	5	5	5	5	5

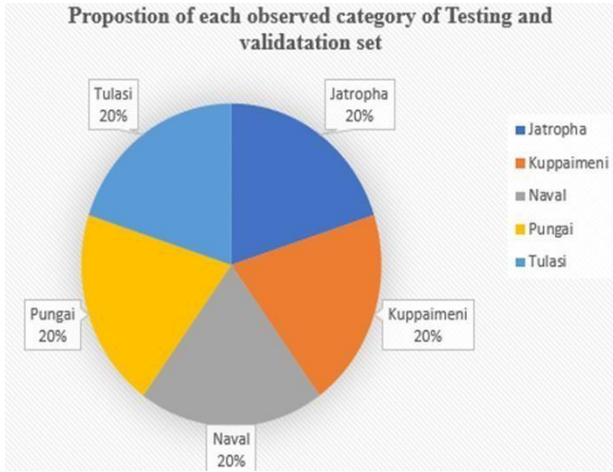


Fig.4. The proportion of samples in train set

Table.2. The summary of the final constructed model

Total number of layers	Non-Trainable layers	Trainable layers
314	249	65

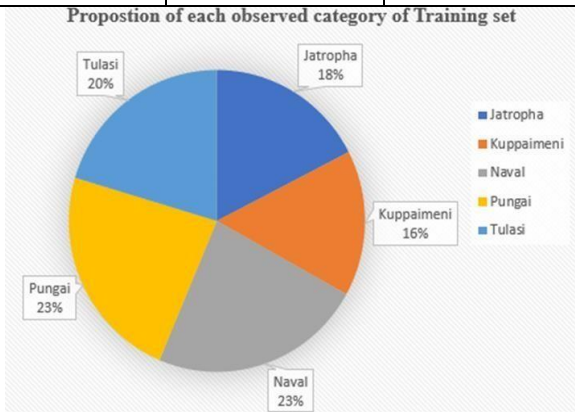


Fig.5. The proportion of samples in test, and validation set

In this project, we achieved 100% test accuracy on the InceptV3 model. Chart.3 displays the detailed result of all epochs. The first 2 column includes training loss and training accuracy while the rest 2 column includes validation loss and validation accuracy. Finally, we achieved 100% accuracy.

Table.3. Training and Validation result

epoch	Train loss	Train accuracy	Val loss	Val accuracy
1	1.4994	0.4218	0.4378	0.9348
2	0.4605	0.8592	0.1888	0.9459
3	0.2046	0.9593	0.1158	0.9612
4	0.1588	0.9812	0.0869	0.9667

Table.4. Testing result

Final loss	0.102
Final accuracy	0.9667



Fig.6. Performance Analysis

Here Series 1, 2, 3 and 4 are represent the Epoch 1, 2, 3 and 4 respectively

Conclusion

From the implementation phase and result, we conclude that: This CNN model is not generic for all different types of plant species. It is just limited to only four different classes. We have developed a model for just only of five different plant Species, namely Pungai, Basil, Kuppaimeni, Jamun, Jatropha curcas. Here the index values are arranged in the order of [0,1,2,3,4] respectively for the above cited classes labels. Accuracy 96.67%, which a good result.

Future Scope

The current work is only for 4 different classes, and our future work is to increase the number of classes with also having a good accuracy in different heterogeneous data. This only can be achieved by modifying the architecture to meet the point cited above. The most difficult thing is that we should not increase the depth of our model, at times it could led to over fitting problem, this is a common problem which arises often in NN. So we have to keep a eye on this while developing a new architecture

References**Journals**

- [1]. K.P. Ferentinos, "Deep learning models for plant disease detection and diagnosis", *Computers and Electronics in Agriculture*, vol. 145, pp. 311-318, 2018.
- [2]. Zhu, M. Zhu and H. Ren, "Method of plant leaf recognition based on improved deep convolutional neural network", *Cognitive Systems Research*, vol. 52, pp. 223- 233, 2018.
- [3]. Y. Uprety, H. Asselin, A. Dhakal, and N. Julien, "Traditional use of medicinal plants in the boreal forest of Canada: review and perspectives," *Journal of ethnobiology and ethnomedicine*, vol. 8, no. 1, p. 7, 2012.

Book

- [4]. Y. Sun, Y. Liu, G. Wang, and H. Zhang, "Deep learning for plant identification in natural environment," *Computational intelligence and neuroscience*, vol. 2017, 2017.
- [5]. P. Barre, B. C. Stover, K. F. Müller, and V. Steinhage, "Leafnet: A computer vision system for automatic plant species identification," *Ecological Informatics*, vol. 40, pp. 50– 56, 2017.
- [6]. G.-h. Song, X.-g. Jin, G.-l. Chen, and Y. Nie, "Two-level hierarchical feature learning for image classification," *Frontiers of Information Technology & Electronic Engineering*, vol. 17, no. 9, pp. 897–906, 2016.

Chapter in a Book

- [8]. K.P. Ferentinos, "Deep learning models for plant disease detection and diagnosis", *Computers and Electronics in Agriculture*, vol. 145, pp. 311-318, 2018.
- [9]. S.H. Lee, C.S. Chan, S.J. Mayo and P. Remagnino, "How deep learning extracts and learns leaf features for plant classification", *Pattern Recognition*, vol. 71, pp. 1-13, 2017.

Conference Proceedings

- [10]. J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255, IEEE, 2009.
- [11]. D. Ezzat, A. E. Hassanien, M. H. N. Taha, S. Bhattacharyya, and S. Vaclav, "Transfer learning with a fine-tuned CNN model for

classifying augmented natural images," in *International Conference on Innovative Computing and Communications*, pp. 843–856, Springer, 2020.