International Journal of Scientific Research in Science, Engineering and Technology
Print ISSN: 2395-1990 | Online ISSN : 2394-4099 (www.ijsrset.com)
doi : https://doi.org/10.32628/IJSRSET

Convolutional Neural Network based methodology for Plant Leaf Classification

Dr. Archana Potnurwar, Jyotsna Kale, Chaitanya Ganvir, Piyush Mahajan, Arsh Motghare, Ritik Dongre

Department of Information Technology, Priyadarshini College of Engineering, Nagpur, Maharashtra, India

Article Info
Volume 9, Issue 3

Page Number : 336-342

Publication Issue :
May-June-2022

Article History
Accepted : 15 May 2022
Published: 30 May 2022

ABSTRACT

A network model that is frequently used in image classification, target recognition, and other domains is known as a convolutional neural network (CNN). This type of neural network structure is considered to be particularly essential in the field of deep learning. In botany, the classification of leaves and the ability to recognise them is particularly significant for recognising new or rare species of trees. Plants are found in almost every part of nature, and their continued existence and development are crucial to the continued existence of all living organisms on earth. The study of the evolutionary law of plants, the preservation of plant species, and the expansion of agricultural practises can all benefit greatly from the identification of species by their leaves, as can the development of agriculture. It is possible to realise the automatic extraction of leaf image features, reduce tedious labour costs, and realise the use of artificial intelligence to classify leaves with the help of this paper, which uses a convolutional neural network in artificial intelligence to identify the leaves of several different kinds of trees that were collected by the Kunming Institute of Botany in Yunnan Province. This paper provides an auxiliary means of artificial intelligence for the study of botany through its contribution to the field of artificial intelligence.

Keywords : Convolutional Neural Network, Image Classification, Target Recognition

I. INTRODUCTION

The convolutional neural network, also known as CNN, is a type of feed forward neural network. It is one of the most important algorithms in machine learning. CNNs feature a convolution layer with a convolution calculation function and the depth architecture of multilayer neural networks. The rapid application of artificial intelligence and pattern recognition has resulted in the obvious benefits of convolutional neural networks in image classification becoming more and more prominent. This has resulted in its widespread application in industry, agriculture, the military, and other fields, including botany. In nature, plants can be found in a broad variety of locations. The first step in moving on with the re-search of botany is to begin with the classification of plant species. Beginning with the size and shape of the leaf to identify the type of tree, this is of considerable value for the research of the tree species itself, as well as the study of the ecological environment of the region, as well as other research work performed by plant researchers. The usage of natural resources and the protection of such resources are dependent on accurate information regarding the types of species and their spatial distribution. Researchers from the Institute of Botany frequently travel to various locations outside of the institute to gather plant leaves. The manual identification of plants is the primary method utilised in traditional plant recognition [1]. The amount of experience that collectors have is quite high [2], primarily as a result of the hostile field environment and the problematic communication. There is a significant reduction in the accuracy in leaf recognition when manual recognition is used [3, 4]. In this study, the Kunming Institute of Botany in Yunnan Province provided a huge number of leaf samples, and an algorithm called convolution neural network was used to determine which of a variety of typical tree species each sample belonged to. A convolutional neural network may accomplish automatic extraction of leaf image attributes, thereby reducing tiresome labour expenses, realising an artificial intelligence approach to classify leaves, and providing aid for botany study using artificial intelligence.

Plant researchers in Kunming, Yunnan, were responsible for the collection of the sample set that was used in this article. The vast majority of leaves are gathered from distant mountain forests both in the United States and internationally. Tower cranes are responsible for gathering them, and then they are scanned into computerised photographs. Following the acquisition of images, OpenCV will be utilised for image preprocessing to build a standardised and uniform sample set. This preprocessing will include denoising, adjusting the image size, and other similar tasks. At the same time, the binarization operation of the leaves will be carried out, which will make it easier to calculate the leaf area, automatically calculate the leaf area index, transpiration rate, photosynthetic rate, and other rates, and improve the efficiency of the working process. During the experiment, as indicated in Figure 1, five leaves were chosen to serve as representatives for the purpose of classification and recognition. Refer to Table 1 for an explanation of the various folders and the types of trees that correspond to them when storing five leaves.

Fig. 1 Five kinds of leaves

Table. 1 Folders correspond to tree species

Folder	**varieties of trees**
Folder 1	Ficus langkokensis Drake
Folder 2	Diospyros xishuangbannaensis
Folder 3	Pometia pinnata
Folder 4	Canarium subulatum Guill.
Folder 5	Parashorea chinensis Wang Hsie

II. Introduction to Convolutional Neural Networks

CNN was conceptualised on the basis of the receptive field. Through an experimental examination of the visual cortex cells of cats in the 1960s, Hubel and colleagues established the rule of the size of the visual receptive field and provided a concise explanation of the idea of the receptive field. The feature map tensor is created when the input image tensor exits the CNN after passing through each layer of neurons and being transformed into the convolutional feature tensor. The term for the portion of the original picture that corresponds in size to each pixel value just on feature map is the receptive field. To put it another way, the pixel values that are contained within a particular region of the input tensor play a role in determining the pixel value of a particular point on the output tensor. Additionally, the pixel value of a particular point on the version was developed is influenced by all of the pixels that are contained within the receptive field region of the input image.

The CNN is a specialised form of the deep neural network model that possesses these three qualities. On only one hand, its neural synapses are only partially coupled to one another. The pixel value of the output feature picture is not calculated from all of the pixel points of the input image (this process is referred to as full connection), but rather it is impacted by the pixels in the responsive field area (called incomplete connection). However, overall impact of adjacent pixels within the receptive field of the input image just on corresponding pixels of the output feature map is not always consistent throughout the entirety of the image. For instance, it is possible that the centre influence is strong while the surrounding influence is minor. As a result, it is required to give weight to both the effect of each pixel position in the receptive field whenever the influence range is indicated by the receptive field. When the value is higher, the amount of influence is also higher. The convolution kernel in CNN [5] is responsible for assigning a weight of effect to each pixel that is part of the receptive field.

On the other hand, the second trait is the fact that neurons that are part of the same layer of the CNN share the same weights. To put it another way, the same convolution kernel is applied to the various receptive fields of the input image in order to construct each pixel of the feature map. Learning how to train and use a deep neural network structure might be challenging. The complexity of the machine learning algorithm is directly proportional to the amount of neuron weights that are used. Reducing the former can help alleviate some of the challenges associated with training, at least to some extent.

CNN is better suited for image language understanding interpretation than the fully connected network because of its network structure, which consists of imperfect connections and weight sharing. This structure imitates and approximates the structure of a true biological neural network.

The third advantage of CNN is that it is able to retrieve the convolution kernel parameters through generating accurate learning, which allows it to circumvent the complicated process of artificial feature extraction. For instance, conventional apps for iris detection and classification like this one need the user to manually extract information such as the length and width of various types of iris flowers, and then categorise iris flowers in accordance with these manually extracted features. There are variations in the quality of physically extracted features, such as whether or not the features extracted are closely related to the outcomes of the prediction, and the process of feature extraction will have an effect on the results of the prediction. The process of manually extracting features is now handled by the convolution kernel in the CNN network instead of being done by hand. The weights of the convolution kernel are not manually assigned; rather, they are randomly assigned when the kernel is initialised. When the total amount of time spent exercising is increased, the optimal weights can be achieved by continued adjustment.

III. Neural model building

In this paper, the building, training, and prediction of a CNN neural network are carried out with the assistance of the machine learning framework from Google called TensorFlow as well as Keras. After the release of TensorFlow2.0 version 2.0, Keras became the advanced deep learning API that was utilised. Keras is a sophisticated neural network application programming interface (API) that was created in Python and is compatible with TensorFlow as a back end. The development of Keras is centred on providing assistance for running experiments quickly. It may quickly transform concepts into the outcomes of experiments with very little loss of time.

OpenCV is utilised in advance to preprocess the image in preparation for the use of a neural network. This preprocessing includes re-moving noise and unifying size. At the same time, the binarization operation of the leaves will be carried out, which will make it easier to calculate the leaf area, automatically calculate the leaf area index, transpiration rate, photosynthetic rate, and other rates, and improve the efficiency of the working process. After that, you should construct a convolutional neural network. Flattening layers and many fully linked layers typically come after several groups of convolution layers and maximum pooling layers in a typical implementation. The convolution procedure is what makes the convolution layer's feature extraction of the image a reality. The number of convolution kernels has no effect on the overall size of the image, but it does have an effect on the number of channels that make up the image tensor. The relu function can be used to activate the convolution feature graph, which adds non-linear features to the linear convolution neural network. This improves the classification curve's ability to differentiate between the various categories. The max-pooling layer retains only the pixels with the highest possible value in the surrounding area. This helps to reduce the overall size of the picture while maintaining the image's most important details. Because of the saved computer resources, the convolution layer may be able to extract a greater number of features. As a result, the greater the number of convolution layers, the greater the number of convolution kernels.

Following a number of rounds of convolution and max-pooling, typically three rounds, followed by flattening layer, the objective is to flatten the picture into a one-dimensional matrix in order to get it ready for multiplication with the matrix of the next fully connected layer. There is the possibility of two layers that are completely joined. It is important to remember that the number of classifications that need to be identified must be equal to the number of neuron matrix columns that are present in the final layer that is fully connected. As can be seen in Figure 2, the topology of the full convolution neural network has been substantially finished up until this point. In most cases, shedding components and regularisation are introduced to the entire neural network in order to prevent the neural network from over-fitting to the training sample set. This is done to prevent the network from producing inaccurate results.

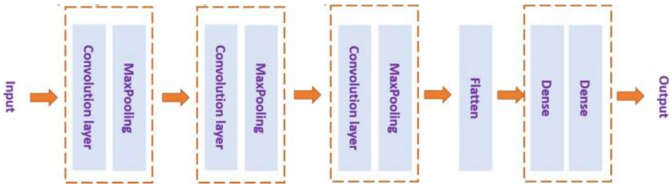

Fig. 2 Neural network model structure

The issue of over-fitting typically arises when there are an excessive number of variables to consider (features). In this particular scenario, the trained equation always provides a good fit to the training data, which means that the loss function is either exactly 0 or extremely near to 0. However, because this curve attempts to fit the training data in whatever way possible, it is unable to generalise to new data samples, which means that new samples cannot be predicted. Because of this, new samples cannot be predicted. Over-fitting occurs when there are an excessive number of variables (features) but not enough training data. Therefore, dropout and regularisation are implemented in order to remedy the problem of over-fitting.

The idea behind Dropout is that the problem of overfitting can be obviously mitigated in each training batch by omitting some feature detectors. This is the core of the Dropout algorithm (letting some hidden layer nodes have values of 0). Let the activation value of a neuron stop working with a given probability p when propagating forward. This can help the model become more generalised because it does not depend as much on certain local characteristics.

In machine learning, regularisation is a useful technique for minimising the risk of model over-fitting and maximising the likelihood of successful generalisation by keeping the complexity of the models under control. The loss function of the model is modified by the addition of an index that describes the complexity of the model. L2 regularisation is the most typical function that is utilised in order to characterise the complexity of the model. Because the weight is being restricted, the model will not be able to arbitrarily accommodate the random noise that is present in the training data. The little weight is reduced when the weight is squared, while the large weight is increased. This has the effect of feature selection, but it does not cause the weight to become 0. Assuming that the parameter of the network layer that needs to be regularised is W, the formula for l2 regularisation looks like this: Formula (1). Ein is the training sample error without the regularisation term, controls the size of the regularisation term, and a larger value of will constrain the performance of the system to a greater extent; or vice versa, Dallas to the auditorium. Among these, Ein is the training sample error without the regularisation term. In actual practise, the standard term is typically added to the goal function, also known as the loss function, and the error of the entire optimization problem is propagated back. This allows the regular term to impact and direct the training of the network. In the field of deep learning, L2 regularisation is more frequently known as "weight attenuation."

$$\boldsymbol{L} = E\text{in} + \lambda \sum \boldsymbol{j}\ \boldsymbol{W2} \tag{1}$$

Behind the most recent full connection layer, an additional Softmax activation unit is installed for good measure. It maps the values of output neurons to the range from 0 to 1, and the aggregate is guaranteed to be 1 after normalisation. This ensures that the sum of the probabilities of several classifications is exactly 1, or one hundred percent.

IV. Sample pretreatment and model training

All of the leaf images are scanned using a scanner, and OpenCV must perform some basic image preprocessing on them. This includes resizing the images so that they are all the same size, filtering them to eliminate noise, and so on. The processed picture samples are then subjected to a random scrambling procedure before being separated into a training set and a test set according to the ratio of 80 percent to 20 percent. The model is responsible for the training of the model. fit () approach is used in the TensorFlow keras database. Adam algorithm is chosen to be the optimizer. The loss is determined by cross entropy. Performance is rated based on classification accuracy.

Cross entropy is able to appropriately describe the difference between the predicted value and the label value when it is used as the basis of loss. The value of the cross entropy increases in proportion to the size of the difference between the two systems, and vice versa. Second, it can be seen from the method of computation that the cross entropy is exclusively related to the label category, and the better the results are, the closer the predicted value that belongs to this category is to 1 (which is the optimal value). The mean square error, which is related not only to the real classification but also to other items, requires that the difference between the real observed and predicted values of all categories be squared again. This is because the mean square error is related to not only the real category but also to other items. If the values that are projected to belong to each group are more average, then the mean square error will be reduced to its lowest possible value. The difficulty of forecasting specific values is addressed by the regression problem, which is distinct from the categorization problem. For instance, forecasting the price of property and the volume of sales both fall under the category of regression questions. It is not a predetermined category that needs to be predicted by these difficulties; rather, it is an arbitrary real number. In most cases, neural networks designed to solve regression problems have only one output node, and the value of this node's output represents the value that is to be predicted. When dealing with regression issues, the mean square error is the loss function that is utilised most frequently. Regarding the cross entropy loss, due to the fact that it is challenging to quantify the intricate similarity matrix that exists between

categories, we are only able to focus on the category to which the sample belongs. This is the case so long as the predicted value of the real category is closer to 1, indicating that it is the more reasonable of the two.
Additionally, the model includes the incorporation of training iteration times. fit () technique. Following the training, the loss curve and accuracy curve may be viewed through the TensorBoard visualisation panel. This makes it possible to easily alter the model's super-parameters. When the training of the model is finished, the weights of the neural network are fixed, and at this point, the model is ready to be utilised for picture prediction and the prediction of leaf species.
It is required to transplant the trained artificial neural network model to the mobile phone terminal in order to make the implementation of the programme more portable and convenient. This can be accomplished by transferring the model to the mobile phone. TensorFlow pro- vides TensorFlow Lite was designed specifically for mobile terminals. Freezing the trained model allows for the model to be converted into the format that is required by TensorFlow Lite. The TensorFlow Lite loading model can then be imported and configured on an Android mobile phone, allowing for the leaf shape prediction function to be added to an Android mobile phone.
The particular step in the process is to save the transformed model file in the assets folder of Android Studio. Android Studio needs to be configured so that it can import TensorFlow Lite. The model should then be called in AndroidStudio. Through the TensorFlowInferenceInterface class, the TensorFlow Java API provides access to all of the necessary functions. In order to use the model, you must first load the libtensorflow inference.so package and then initialise the object known as the TensorFlowInferenceInterface. The image is then transmitted to the neural network model by utilising the inference Interface object, which includes calling functions such as feed (), run (), and fetch (), amongst other functions, and the prediction result is obtained. On the mobile phone side, it is also important to preprocess the images of leaves that are stored in the mobile device's photo or picture library.
TensorFlow Lite makes it possible to execute artificial intelligence classification and recognition on Android mobile terminals. It also provides plant researchers and hobbyists with auxiliary tools that are more convenient and versatile.

V. Conclusion

Image classification is one of the most common applications for deep learning's convolutional neural network, which has vital applications across a variety of industries and has the potential to boost classification efficiency significantly. When it comes to botany, the classification and naming of leaves are extremely significant steps in the process of recognising new or uncommon species of trees. The accurate acquisition of information regarding the types of species as well as their spatial distribution is the foundation for the utilisation and protection of natural resources [6]. The identification of species is of great assistance to plant evolution research, the protection of plant species, and the development of agricultural practises. In order to gather plant leaves, researchers from the Institute of Botany frequently venture out into the field. Traditional methods of identifying plants are mostly based on manual recognition, which is constrained by factors such as environment and communication, and necessitates a very high level of skill on the part of collectors. The level to which accuracy may be achieved through manual recognition of leaves is somewhat limited. In this particular piece of research, a convolutional neural network technique is employed to classify the various types of leaves. A convolutional neural network may accomplish automatic extraction of leaf image attributes, thereby reducing tiresome labour expenses, realising an artificial intelligence approach to classify leaves, and providing aid for botany study using artificial intelligence. OpenCV is employed for the picture preprocessing step, and then TensorFlow and Keras are utilised for the model construction, training, and prediction phases of the leaf classification process. When combined with TensorFlow Lite, the model loading and prediction are executed on an Android mobile phone terminal. This not only increases the portability of the application but also makes it possible to deliver an artificial intelligence auxiliary tool that is both practical and speedy for use in plant research.

VI. REFERENCES

[1]. Shuwei Wei, Shangyou Zeng, Yue Zhou, Xinjiao Wang. (2020) Tree Leaves Classification Based on Deep Convolution Neural Network. Modern electronic technology, 09 96-100.

[2]. Muhammad Azfar Firdaus Azlah, Lee Suan Chua, Fakhrul Razan Rahmad, Farah Izana Abdullah, Sharifah Rafidah Wan Alwi. (2019) Review on Techniques for Plant Leaf Classification and Recognition. Computers, Volume 8, Issue 4.

[3]. Febri Liantoni, Rifki Indra Perwira, Syahri Muharom, Riza Agung Firmansyah, Akhmad Fahruzi. (2019) Leaf classification with improved image feature based on the seven moment invariant.

[4]. Xiumei Zhang, Yaqiu Liu. (2006) Recognition system of leaf images based on neuronal network. Journal of Forestry Research, Page 243-246+266.

[5]. Muammer Turkoglu, Davut Hanbay. (2019) Recognition of plant leaves: An approach with hybrid features produced by dividing leaf images into two and four parts. Applied Mathematics and Computation, Volume 352, 2019. PP 1-14.

[6]. K. Pankaja,V. Suma.(2020)Plant Leaf Recognition and Classification Based on the Whale Optimization Algorithm (WOA) and Random Forest (RF).Journal of The Institution of Engineers (India): Series B, Volume 101, 2020. PP 597-607.

[7]. Kaya, A., Keceli, A.S., Catal, C., Yalic, H.Y., Temucin, H. and Tekinerdogan, B, "Anal- ysis of transfer learning for deep neural network-based plant classification models." Computers and Electronics in Agriculture, Vol. 158, March, 2019.

[8]. Barbedo, J.G, "Factors influencing the use of deep learning for plant disease recog- nition", Biosystems engineering, Vol. 172, May,2018.

[9]. Ghazi, M.M., Yanikoglu, B. and Aptoula, E, "Plant identification using deep neural net- works via optimization of transfer learning parameters", Neurocomputing, Vol. 235, April, 2017.

[10]. Lee, S.H., Chan, C.S., Mayo, S.J. and Re- magnino, P., "How deep learning extracts and learns leaf features for plant classifi- cation", Pattern Recognition, Vol. no: 71, May, 2017.

[11]. Barbedo, J.G.A., "Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant dis- ease classification", Computers and Elec- tronics in Agriculture, Vol. 153, Oct, 2018.

[12]. Zhu, X., Zhu, M. and Ren, H., "Method of plant leaf recognition based on improved deep convolutional neural network", Cog- nitive Systems Research, Vol. 52, Dec, 2018.

[13]. Garcia-Garcia, A., Orts-Escolano, S., Oprea, S., Villena-Martinez, V., Martinez- Gonzalez, P. and Garcia-Rodriguez, J., "A survey on deep learning techniques for im- age and video semantic segmentation, Ap- plied Soft Computing", Vol. 70, May, 2018.

[14]. Noon, S.K., Amjad, M., Qureshi, M.A., Mannan, A., "Use of deep learning tech- niques for identification of plant leaf stresses: A review", Sustainable Comput- ing: Informatics and Systems, Vol.28, Dec, 2020.

Cite this article as :

Dr. Archana Potnurwar, Jyotsna Kale, Chaitanya Ganvir, Piyush Mahajan, Arsh Motghare, Ritik Dongre, "Convolutional Neural Network based methodology for Plant Leaf Classification", International Journal of Scientific Research in Science, Engineering and Technology (IJSRSET), Online ISSN : 2394-4099, Print ISSN : 2395-1990, Volume 9 Issue 3, pp. 336-342, May-June 2022.

Journal URL : https://ijsrset.com/IJSRSET2293128