Journal of Forests
2021 Vol. 8, No. 1, pp. 61-70.
ISSN(e): 2409-3807
ISSN(p): 2413-8398
DOI: 10.18488/journal.101.2021.81.61.70
© 2021 Conscientia Beam. All Rights Reserved.

CLASSIFICATION AND RECOGNITION OF URBAN TREE DEFECTS IN A SMALL DATASET USING CONVOLUTIONAL NEURAL NETWORK, RESNET-50 ARCHITECTURE, AND DATA AUGMENTATION

Arjun Dixit[1+]
Yeong Nain Chi[2]

[1,2]Department of Agriculture, Food and Resource Sciences, University of Maryland Eastern Shore Princess Ann, MD, USA.
[1]Email: adixit@umes.edu Tel: +1-443-985-8127
[2]Email: ychi@umes.edu Tel: +1-410-651-8186

(+ Corresponding author)

Article History
Received: 22 January 2021
Revised: 16 February 2021
Accepted: 5 March 2021
Published: 8 April 2021

Keywords
Convolutional neural network
MATLAB
Image classification
Urban tree defect identification
Deep learning
ResNet-50
Small dataset
Data augmentation.

ABSTRACT

Identifying hazard trees in urban setup is a time-consuming and tedious task and therefore concerned organizations and homeowner associations may not identify and fix such hazard trees in time. The purpose of this study was to identify the type of defects in the trees with the use of convolutional neural networks. This technology could speed up the process of identifying hazard trees. The study used the Image Processing Toolbox of MATLAB 2019a to process and classifies the images into one of the seven most prominent types of tree defects. The CNN used for this classification was ResNet-50. The Tree Defects dataset was prepared from images from publicly available sources. Further, the accuracy of the classification of these images into each of the defect categories was tested by obtaining a confusion matrix. The performance of ResNet-50 architecture was compared on three more publicly available and common research datasets Caltech101, Flower, and Dogs. The novel Tree Defects dataset was very small and had only 298 images. For its effectiveness on smaller datasets, ResNet-50 architecture was used along with data augmentation of tree defects images by rotating them 90-degrees clockwise and anti-clockwise. The effect of the proportion of the training dataset on model performance was also evaluated by training the model on 70%, 80%, and 90% of the total images in the dataset. The augmented Tree Defects dataset had 894 images. The model performance improves by 43.56% on the augmented Tree Defects dataset. The augmented model achieved the highest classification accuracy of 91.48%.

Contribution/Originality: This study is one of the very few studies that have investigated the ways to find an image classification model that delivers high accuracy on smaller datasets. Mostly, a machine learning model is believed to perform better on vast datasets but building large datasets is costly and time-consuming.

1. INTRODUCTION

Trees are an integral part of urban society. Their presence in urban infrastructure is necessary for environmental and aesthetic reasons. They help to keep the balance of oxygen and carbon dioxide in the air by absorbing the latter and releasing the former into the environment in return. Therefore, every urban development has special provisions planned for trees and other greenery. In addition to the environmental benefits, they also add to the beauty of urban development. But due to natural or external forces, many trees develop defects that make them weak and turn them into a hazard. A tree becomes a "hazard tree" when it has structural defects that could lead them to damage the property and lives. Therefore, timely management of hazard trees can save them.

Managing a large number of trees in an urban area and tracking their health would take a lot of time and resources. Often, a hazardous tree would already cause some damage before being acted upon. Therefore, a systematic tracking and classification model could solve the problem of identification of tree defects that could make a tree hazardous. Lately, the use of artificial intelligence and deep learning methods to solve complex problems related to image data has produced encouraging results. These models have been found equally effective and accurate in different fields of application.

In their research, In past research, models have been trained using NVIDIA DIGITS deep learning training system [1]. The researchers in this study used a convolutional neural network deep learning model to automatically classify various diseases and nutritional deficiencies on apple trees based on a dataset of images. The classification results of the model were found to be 97.3% accurate, more than the accuracy of the human experts in that field. Six different kinds of plant diseases were automatically classified using a Multilayer Perceptron (MLP) model that achieved a classification accuracy as high as 94% [2]. A similar method of classification was used to classify healthy and unhealthy melons [3]. This study used a Convolutional Neural Network (CNN) model to classify images of melons as healthy and unhealthy. The overall classification results were found up to 97.5% accurate. CNN's have also been used to classify wood defects in the past.

Wood defects have also been with the use of deep convolutional neural networks. The study also showed how the performance of a classification model can be improved using data augmentation [4]. Their classification accuracy increased from 87.0% to 99.13% with the use of data augmentation and other optimization. Studies have also applied CNN-based computer vision techniques to classify urban environments of different cities using their geospatial images. A study also found that ResNet-50 architecture produced better results than VGG-16 architecture consistently. Research work from the past proves that convolutional neural networks generate good results and can handle a large amount of image data for pre-processing, computation, and image classification [5].

The purpose of this study is to explore the application of data augmentation to improve the efficiency of smaller datasets. Most of the neural networks produce good results when the input data are large. The performance of these models deteriorates dramatically as the size of the input dataset becomes smaller. It is often not feasible for organizations and researchers to construct a large input dataset for their analysis due to cost and availability issues. Therefore, a method to improve the model performance on smaller datasets is needed. In cases where data are not large, the size of the input dataset is increased artificially using data augmentation. This method increases the performance of the model significantly and seems to work well on smaller datasets. There is a lack of quality datasets related to urban trees and their planning. Therefore, exploring the application of data augmentation and specific architecture models on smaller data sets to identify tree defects would help in obtaining accurate results and thus, help in the management of urban trees by improving tree defect identification.

2. CONVOLUTIONAL NEURAL NETWORKS

The CNNs were specially designed to work on images. They can handle image classification with good accuracy. They also work well with non-image data. The convolutional neural networks have different layers in their architecture. They consist of the convolution layer, the ReLU layer, pooling layers, and a fully connected layer. The flow of input and its progress through various layers is shown in Figure 1.

A convolutional neural network extracts and learns the features of an image and then classifies it as one of the predefined classes. Deep convolutional neural networks can be trained rapidly with the help of a GPU (Graphics Processing Unit). A CNN starts with breaking down the input image into small pixels. The pixels are represented as 2D or 3D according to the color of the image. If the image is black and white, the pixel array is 2D. A pixel value lies between 0 and 255 where 0 is black and 255 is completely white. A 3D array of the colored image has three color layers, an RGB layer. RGB stands for Red, Blue, and Green, respectively. Each color has a value from 0 to 255. Every color tone can be found with the combination of these three colors.

© 2021 Conscientia Beam. All Rights Reserved.

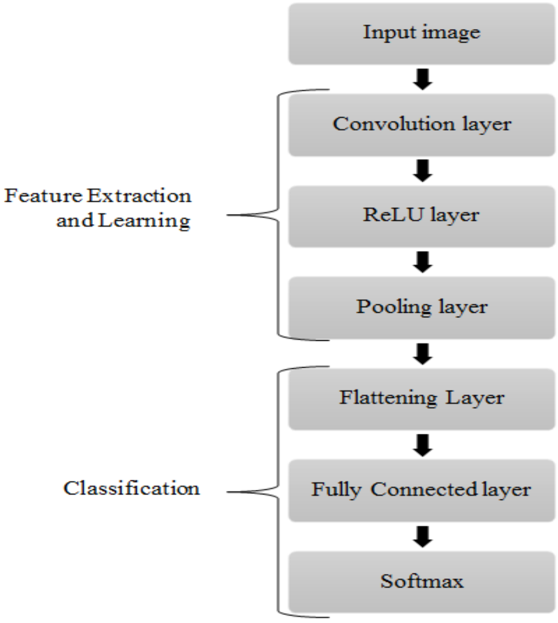


Figure-1. Layers of a Convolutional Neural Network.

A convolutional layer extracts the features of the input image. It extracts the low-level features of an image such as edges, corners, etc. from the input image. It further extracts higher-level features. A kernel or filter is used to carry out the convolutional operations on the image by which it extracts the useful features. A kernel is a collection of pixels that move across the image pixels one after the other to analyze every pixel of the image. The kernel can move across the image at a fixed distance at a time. This distance is called stride. In this process, it crosses the entire image. There can be more than one convolutional layer depending upon the complexity of the image. The kernel moves along a linear path. It learns the linear computations as its output as it moves across the image after each stride. In reality, the images a CNN has to deal with are highly non-linear. Therefore, a non-linear activation function is added to this layer. This is done by the ReLU layer or the Rectified Linear Unit layer. It introduces a non-linear activation function to the network to increase the non-linearity in the system, thus, increasing its classification efficiency and accuracy. Other non-linear activation functions that are commonly used are sigmoid and tanh. ReLU is faster and more accurate.

Once through the convolutional layer and the ReLU layer, the image features are already extracted. The next layer on CNN is the pooling layer. This layer reduces the size of the image and hence, the computational difficulty. In addition to this, a pooling layer also extracts advanced features of the input image such as positional and rotational invariant. Therefore, it adds to the classification efficiency and accuracy of the model. It also reduces over-fitting. A max-pooling returns the maximum of the values covered by the kernel while the average pooling returns the average value. Max pooling is always a better choice as it suppresses noise in the network too. By this time, the model successfully learns the features, both the low level e.g. edges, etc., and high level e.g. rotational invariant, etc.

After obtaining a pooled feature map of the input image, next comes the flattening layer. This is the first input layer for the classifier model. It obtains the pooled feature map from the pooling layer as its input and converts it into a column like a flattened 1D array. This flattened array becomes the input for the FC layer or fully connected layer for classification. In the FC layer, an artificial neural network classifies the input image as one of the predefined classes. The artificial neural network combines the input features to generate more information. The FC layer is called so because every neuron in this layer is connected with every neural in the next layer. The FC layer uses a classifier in the output layer that uses a Softmax activation function. CNN is better than other classifiers as it identifies and extracts the image features at the same time. Because of this reason, it is faster than other neural networks [6].

© 2021 Conscientia Beam. All Rights Reserved.

3. MATERIALS

The tree defects dataset was prepared using publicly available images for seven tree defects downloaded from (https://www.forestryimages.org/browse/bareas.cfm?domain=23). The dataset consists of 298 images classified into seven types of defects as shown in Figure 2.

a) Cracks: When the bark of a tree starts to separate, it eventually starts splitting the tree as it gets deeper. If such cracks are visible in the stem of the tree, it could be very dangerous as the tree may split and fall, damaging the property and lives nearby. Even if a branch of a tree has cracks, it could still cause some serious harm.

b) Decay: The tree wood starts decaying due to factors like fungal growth and other microbial activity. This type of defect grows from inside out, therefore, it is not visible unless it reaches the outer surface of the tree. Decay can weaken the tree to a great extent by making the wood soft. Due to this reason, they are unable to hold the weight of the branches and finally collapse.

c) Cankers: This is another disease in a tree that may develop due to microbial and bacterial infection. Trees that are weak in nutrition are more prone to such disease. Unlike decay, it starts from the surface. This is a localized infection on the outer surface of the tree that eats up that area. Resultantly, the stem around the infected area is likely to break. Such trees are more likely to fall in stormy weather.

d) Weak Branch Unions: As the name suggests, the branches of the trees that are not strongly attached to the tree tend to make this defect. In most cases, such branches grow so close to each other that bark may grow in between them. The growth of this bark may push these branches to split apart. This phenomenon can be faster and more dangerous if such branches are upright. They may fall and cause damage.

e) Root Problems: The trees that have weaker roots tend to be the first ones to fall in harsh weather conditions. They may also weaken and fall gradually on a normal day. Root problems consist of root decay, paving-over roots, etc. The root may grow out of the surface of the soil and grow wide to search for nutrition. This can damage nearby development such as roads, underground pipes, and wiring, etc.

f) Deadwood: It is a dead branch of a tree that could fall anytime. Usually, deadwood may be the result of natural factors and physiological changes in the tree. External factors such as insect activity or disease could also trigger this defect.

g) Poor Architecture: This defect arises due to the poor growth of the tree. Trees that have strange shapes may be dangerous at the same time. Such strange shapes and architecture of the trees could be the result of improper pruning, harsh weather conditions such as storms, etc.

In addition to the images available on this website, three other publicly available datasets were used in the study to compare results as follows:

1. Stanford Dogs Dataset: The second dataset used in the study was "Stanford Dogs Dataset" which can be retrieved from (http://vision.stanford.edu/aditya86/ImageNetDogs/). This dataset has 120 categories of dogs. The dataset has 20,580 images in total. It has been built by using images from the ImageNet dataset. It is widely used for studies on image classification [7].

2. Caltech101: Caltech101 is an openly available dataset that has 9,146 images and 101 categories [8]. The dataset was built in 2004 and has been extensively used for research in the field of deep learning, object detection, and image classification. There are 40 to 800 images in each category. However, for research purposes, 1149 images from 17 categories that related to animals, plants, and agriculture were selected. This includes categories like ants, bonsai, sunflower, etc.

3. Flower Dataset: The flower dataset has 3670 images and 5 categories, namely daisy, dandelion, roses, sunflower, and tulip. It is available for public use at the link (https://www.tensorflow.org/tutorials/images/classification).

© 2021 Conscientia Beam. All Rights Reserved.

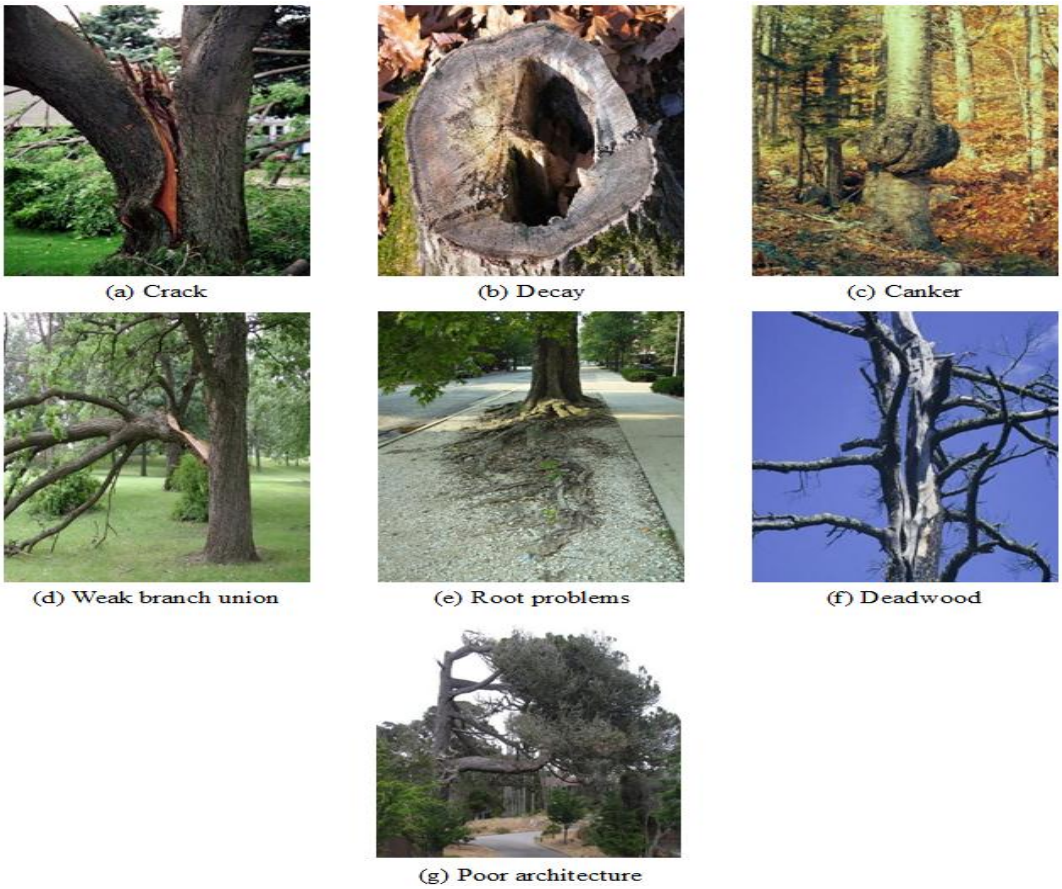


Figure-2. Seven types of tree defects.

4. METHODS

The ResNet-50 architecture was used to develop a CNN model for classification and used on four image datasets taken in this study. ResNet-50 is the residual network architecture that has produced ground-breaking performances in the past not just in image classification, but also in object detection and face recognition. Its architecture has four stages. Stage 1 starts after a 3x3 max-pooling layer. As shown in Figure 3, ResNet-50 does convolutional using 7x7 kernels, and max-pooling at 3x3 kernels size. For each residual function, 3 layers are stacked one on the other. These three layers are 1x1, 3x3, and 1x1 convolution, where the 1x1 layers perform the major responsibility of reducing and then restoring the image, while the 3x3 layer has smaller dimensions. The second stage has four similar residual functions, the third stage has six, and the fourth stage has three residual functions. The output of the last residual function of stage four becomes the input for the average pooling layer that further feeds the input to the fully connected (FC) layer.

"ResNet" stands for "Residual Network". The ResNet-50 architecture proves to be better than the deep convolutional neural networks. This is because it directly fixes the problem of degradation in the deeper convolutional neural networks, or DCNNs. It is a common understanding that increasing the number of layers in a model increases its accuracy and performance. But, as the model becomes deeper, it starts to become less accurate. This happens due to the increase in the error percentage that is caused by the "Vanishing Gradient Effect" during the process of backpropagation. It is because of this problem that the initial layers of a deep model fail to learn and therefore, their weights hardly change, or remain constant. Therefore, the model doesn't learn as the training fails to converge. In a deep network, the output y is given by:

$$y = F(x, \{Wi\}) + x$$

Here, x is the input, and F(x, {Wi}) is the residual block, and Wi is the number of layers in the residual block. A residual block is shown in Figure 4.

© 2021 Conscientia Beam. All Rights Reserved.

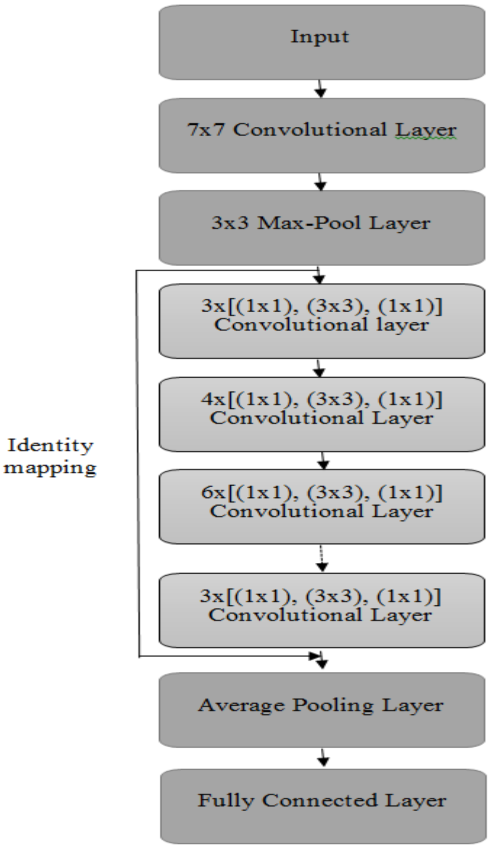

Input
7x7 Convolutional Layer
3x3 Max-Pool Layer
3x[(1x1), (3x3), (1x1)] Convolutional layer
4x[(1x1), (3x3), (1x1)] Convolutional Layer
6x[(1x1), (3x3), (1x1)] Convolutional Layer
3x[(1x1), (3x3), (1x1)] Convolutional Layer
Identity mapping
Average Pooling Layer
Fully Connected Layer

Figure-3. Architecture of ResNet-50.

Source: Kaushik [9]

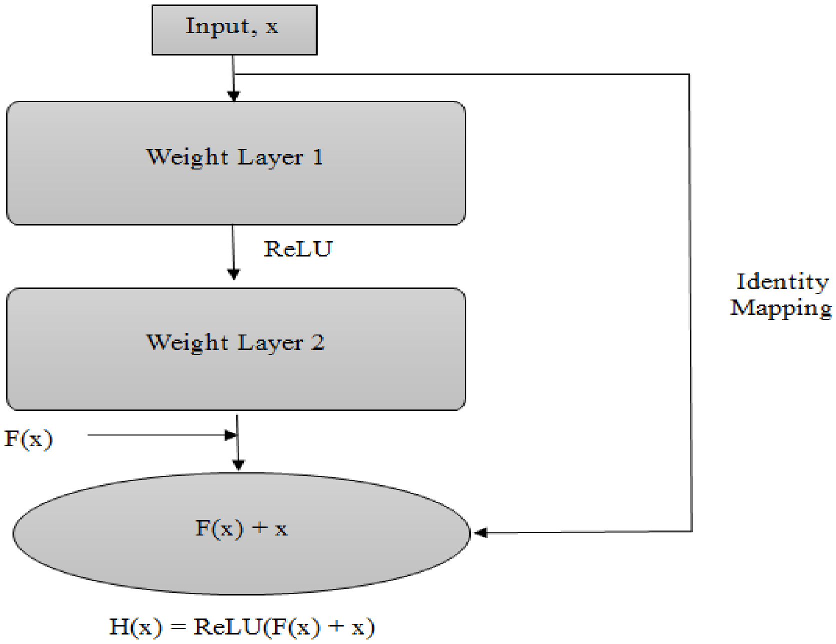


Figure-4. A Residual block.

Source: Nahar, et al. [10].

While in simple deep networks, convolutional, pooling, activation, and fully connected (FC) layers are stacked one on the other, in ResNet-50, an identity mapping is introduced between the layers. An identity mapping to the input gives the output that is the same as the input. A residual network makes use of the residual function which is the difference between the input and the output of the residual block. In Figure 5, gradient path 2 has to go through

© 2021 Conscientia Beam. All Rights Reserved.

weights, while gradient path 1 doesn't have to encounter weights. Therefore, the gradient reaches the input layer without any change in the signal from the output during backpropagation.

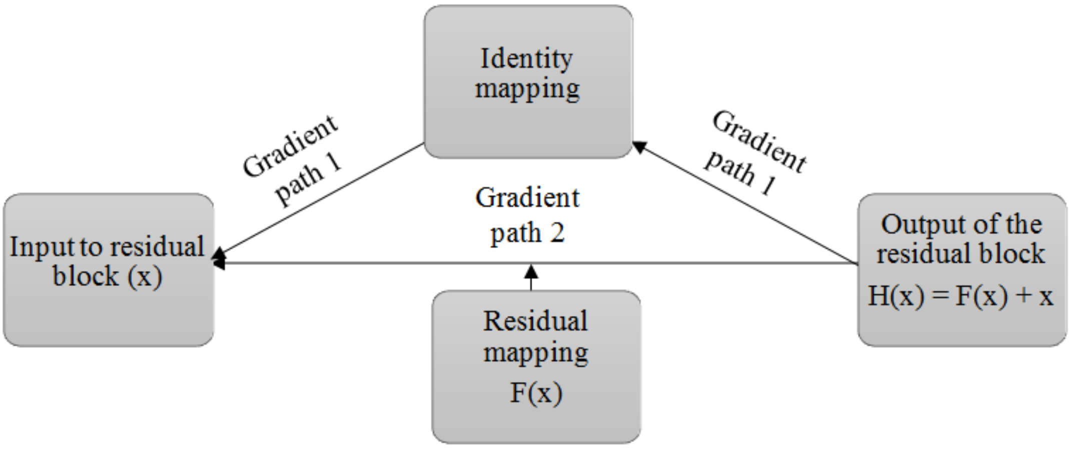


Figure-5. Gradient paths in ResNet-50 network.

Source: Sachan [11].

The image processing toolbox of MATLAB 2019a was used to process the tree defect images dataset and was classified using the ResNet-50 CNN. The ResNet-50 was used in MATLAB by downloading Deep Learning Toolbox Model for ResNet-50 Network. The classification model was run on each of the four datasets and the accuracy percentage was recorded. The partition ratio was changed to see the variation in the results. The two partition ratios of the training set and test set were 70:30 and 85:15, respectively. An additional partition ratio of 90:10 was tried on the Tree Defects dataset. The novel dataset of Tree defects was also augmented by rotating its images 90-degree clockwise and anti-clockwise to see its effect on classification accuracy.

5. RESULTS

Image classification using convolutional neural networks is a supervised learning and classification method. In this method, a fully connected layer at the end of the network classifies the images in the dataset into one of the seven defects mentioned earlier in section IV. First, the effectiveness and accuracy of ResNet-50 architecture were tested on publicly accepted datasets Dogs dataset, Flower dataset, and Caltech101 dataset. The performance of the model on these datasets was compared. A common trend was observed. The classification accuracy showed improvements as the share of the test set was improved.

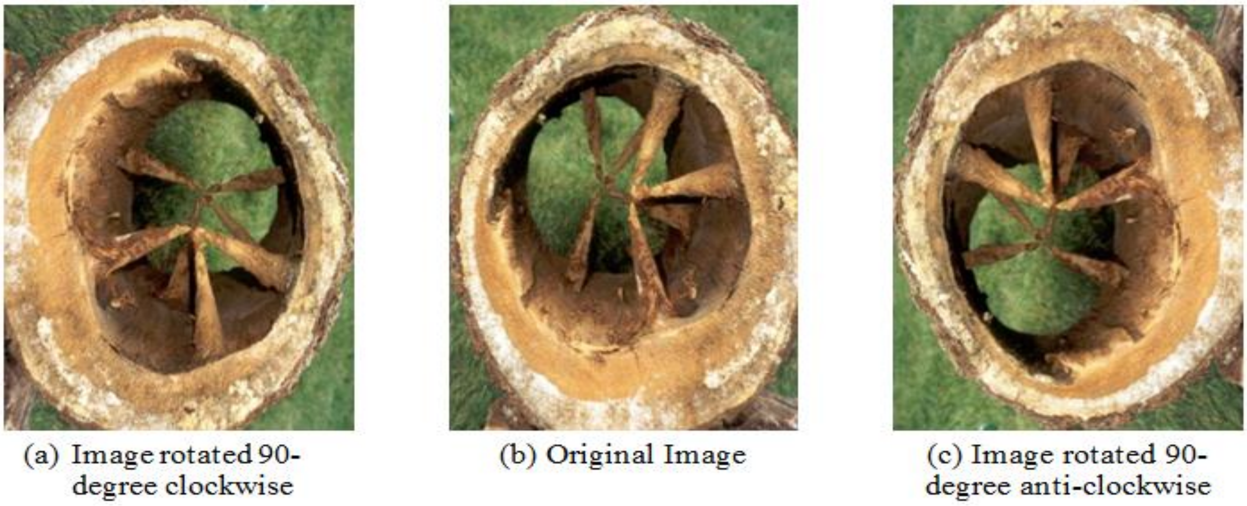
Table 1 shows the classification results on four datasets used in the study. The highest classification accuracy of 93.04% was shown by Caltech101. The accuracy of all three datasets was higher than the 'Tree Defects' dataset, on which the model performed poorly. This could be mainly due to the size of the dataset and the smaller number of images present in each category of tree defects. An additional dataset partition ratio of 90:10 that was tried on the Tree Defects dataset further increased its efficiency than the previous iterations. It was seen that neural networks work better on large datasets and smaller datasets tend to limit the performance of the model. This was seen in the accuracy of the classification of the novel Tree Defects dataset.

Table-1. Results of the accuracy of CNN classification model.

Model Specifications	Dataset			
	Dogs	Flower	Caltech101	Tree Defects
70% training set without data augmentation	76.21%	88.01%	93.04%	39.80%
85% training set without data augmentation	77.85%	89.74%	93.46%	41.02%

© 2021 Conscientia Beam. All Rights Reserved.

Second, the performance of the model on the Tree Defects dataset was poor. Its data was augmented to check if it improved model accuracy. Data augmentation was used to overcome the effect of overfitting. Figure 6. Shows data-augmented image from Tree Defects dataset.


(a) Image rotated 90-degree clockwise

(b) Original Image

(c) Image rotated 90-degree anti-clockwise

Figure-6. Augmented Images from Tree Defects dataset.

Augmentation gave the model additional data to train. This improved its classification accuracy. Images were rotated 90-degrees clockwise and anti-clockwise, respectively. Thus the model had thrice the number of images as the original dataset to train. It was observed that data augmentation brought drastic improvement in the results with the Tree Defects dataset. Table 2 shows the comparison of classification accuracy with and without data augmentation of the Tree Defects dataset at three different proportions of the training set.

Table-2. Effect of data augmentation of Tree Defects dataset on overall classification accuracy.

Data Augmentation	The proportion of Training set		
	70%	85%	90%
Without data augmentation	39.80%	41.02%	47.92%
With data augmentation	77.75%	83.06%	91.48%

The performance of the model saw an encouraging jump in classification accuracy with the application of a simple augmentation method. When the CNN model was trained on rotated images of the same kind, it yielded better results because it could learn better by overcoming its limitation of being rotational invariant.

Being rotational invariant, a CNN model fails to classify an image correctly if it is rotated. Training it on rotated images of the same kind seemed to work well and improved the results significantly, as shown in Table 2. This trend was consistently observed with all three proportions of the training set. The CNN model trained on an augmented training set achieved a maximum classification accuracy of 91.48%.

6. DISCUSSION AND CONCLUSION

The urban tree management and planning need to follow a systematic approach to avoid conflicts with infrastructure development, cost-effectiveness, and safety. Thus, urban organizations need to develop a model that helps in the proactive management of urban trees and any related defects. Managers also do data modeling of costs associated with hazardous trees to decide on removing or maintaining them [12]. A suitable neural network model could improve the efficiency, accuracy, biases, and reproducibility of cost estimation and risk assessment models [13]. The application of neural networks to identify tree defects could expedite the analysis and improve results. Identifying costs associated with urban tree failures is important because such costs are increasing over the years. Urban management bodies in the Netherlands saw a notable increase in the amount they spent as compensation

© 2021 Conscientia Beam. All Rights Reserved.

during the 1960s to 2010 with an average of €2,244 which is worth €2546/$3045 approximately in 2020 [14]. The present value of money was calculated using publicly available online conversion tools.

The ResNet-50 architecture produced encouraging results with Caltech101, Flower, and Dogs dataset. However, it did not produce similar results on a much smaller Tree Defects dataset that only had 298 images. However, the allocation of more images to the training set of the model upped the model performance while classifying images of the Tree Defects dataset. The augmented dataset had 894 images. The CNN model achieved a classification accuracy of 77.45% to 91.48%.

The size of this dataset was still very small compared to other datasets that are used with a CNN model. The performance results after data augmentation established the possibility of using ResNet-50 architecture on small datasets to achieve higher classification accuracy. Variation in the proportion of the training set also enhanced the performance. Applying the simple data augmentation technique of rotating images 90-degrees clockwise and anti-clockwise improved the performance by as much as 43.56%.

This performance may improve further if advanced augmentation methods such as dropout, L2 regularization, etc. are applied. In addition to using better augmentation methods, building a larger dataset could increase the generalization ability of such classification models. Also, the performance of various other pre-trained architectures like ResNet-101, VGG-16, VGG-19, etc. on such smaller datasets could be compared. Deeper research in this field could unfold effective and accurate classification models that could produce breakthrough results and solve complex problems in this field and beyond.

Funding: This work is supported by the USDA National Institute of Food and Agriculture, McIntire Stennis project [Accession No. 1019401].
Competing Interests: The authors declare that they have no competing interests.
Acknowledgement: All authors contributed equally to the conception and design of the study.

REFERENCES

[1] L. G. Nachtigall, R. M. Araujo, and G. R. Nachtigall, "Classification of apple tree disorders using convolutional neural networks," presented at the 2016 IEEE 28th International Conference on Tools with Artificial Intelligence (ICTAI), 2016.

[2] H. Al Hiary, A. S. Bani, M. Reyalat, M. Braik, and Z. ALRahamneh, "Fast and accurate detection and classification of plant diseases," *International Journal of Computer Applications*, vol. 17, pp. 31–38, 2011. Available at: https://doi.org/10.5120/2183-2754.

[3] W. Tan, C. Zhao, and H. Wu, "Intelligent alerting for fruit-melon lesion image based on momentum deep learning," *Multimedia Tools and Applications*, vol. 75, pp. 16741–16761, 2015. Available at: https://doi.org/10.1007/s11042-015-2940-7.

[4] T. He, Y. Liu, Y. Yu, Q. Zhao, and Z. Hu, "Application of deep convolutional neural network on feature extraction and detection of wood defects," *Measurement*, vol. 152, p. 107357, 2020. Available at: https://doi.org/10.1016/j.measurement.2019.107357.

[5] A. Albert, J. Kaur, and M. C. Gonzalez, "Using convolutional networks and satellite imagery to identify patterns in urban environments at a large scale," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2017, pp. 1–10.

[6] M. A. F. Azlah, L. S. Chua, F. R. Rahmad, F. I. Abdullah, and S. R. Wan Alwi, "Review on techniques for plant leaf classification and recognition," *Computers*, vol. 8, p. 77, 2019. Available at: https://doi.org/10.3390/computers8040077.

[7] A. Khosla, J. Nityananda, Y. Bangpeng, and L. Fei-Fei, "Novel dataset for fine-grained image categorization," presented at the First Workshop on Fine-Grained Visual Categorization (FGVC), IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011, 2011.

© 2021 Conscientia Beam. All Rights Reserved.

[8] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," presented at the 2004 Conference on Computer Vision and Pattern Recognition Workshop, 2005.

[9] A. Kaushik, "Understanding ResNet50 architecture. OpenGenus IQ: Learn Computer Science. Retrieved from: https://iq.opengenus.org/resnet50-architecture/," 2020.

[10] P. Nahar, S. Tanwani, and N. S. Chaudhari, "Fingerprint classification using deep neural network model Resnet50," *International Journal of Research and Analytical Reviews*, vol. 5, pp. 1521–1535, 2018.

[11] A. Sachan, "Detailed guide to understand and implement ResNets. CV-Tricks.Com. Retrieved from: https://cv-tricks.com/keras/understand-implement-resnets/," 2019.

[12] J. Vogt, R. J. Hauer, and B. C. Fischer, "The costs of maintaining and not maintaining the urban forest: A review of the urban forestry and arboriculture literature," *Arboriculture & Urban Forestry*, vol. 41, pp. 293-323, 2015.

[13] R. W. Klein, A. K. Koeser, R. J. Hauer, G. Hansen, and F. J. Escobedo, "Risk assessment and risk perception of trees: A review of literature relating to arboriculture and urban forestry," *Arboriculture & Urban Forestry*, vol. 45, pp. 23-33, 2019. Available at: https://doi.org/10.48044/jauf.2019.003.

[14] M. A. van Haaften, M. P. M. Meuwissen, C. Gardebroek, and J. Kopinga, "Trends in financial damage related to urban tree failure in the Netherlands," *Urban Forestry & Urban Greening*, vol. 15, pp. 15–21, 2016. Available at: https://doi.org/10.1016/j.ufug.2015.11.002.

Views and opinions expressed in this article are the views and opinions of the author(s), Journal of Forests shall not be responsible or answerable for any loss, damage or liability etc. caused in relation to/arising out of the use of the content.

© 2021 Conscientia Beam. All Rights Reserved.