Plant Leaf Disease Detection and Classification Based on CNN with LVQ Algorithm

Melike Sardogan
Institute of Science and Engineering
Yalova University
Yalova, Turkey
melikesardogan@gmail.com

Adem Tuncer
Department of Computer Engineering
Yalova University
Yalova, Turkey
adem.tuncer@yalova.edu.tr

Yunus Ozen
Department of Computer Engineering
Yalova University
Yalova, Turkey
yunus.ozen@yalova.edu.tr

***Abstract*—The early detection of diseases is important in agriculture for an efficient crop yield. The bacterial spot, late blight, septoria leaf spot and yellow curved leaf diseases affect the crop quality of tomatoes. Automatic methods for classification of plant diseases also help taking action after detecting the symptoms of leaf diseases. This paper presents a Convolutional Neural Network (CNN) model and Learning Vector Quantization (LVQ) algorithm based method for tomato leaf disease detection and classification. The dataset contains 500 images of tomato leaves with four symptoms of diseases. We have modeled a CNN for automatic feature extraction and classification. Color information is actively used for plant leaf disease researches. In our model, the filters are applied to three channels based on RGB components. The LVQ has been fed with the output feature vector of convolution part for training the network. The experimental results validate that the proposed method effectively recognizes four different types of tomato leaf diseases.**

***Keywords*—Leaf Disease Detection, Leaf Disease Classification, Convolutional Neural Network (CNN), Learning Vector Quantization (LVQ)**

I. Introduction

Plant diseases affect the growth and crop yield of the plants and make social, ecological and economical impacts on agriculture. Recent studies on leaf diseases show how they harm the plants. Plant leaf diseases also cause significant economic losses to farmers. Early detection of the diseases deserve special attention. Plant diseases are studied in the literature, mostly focusing on the biological aspects. They make predictions according to the visible surface of plants and leaves. Detection of diseases as soon as they appear is a vital step for effective disease management. The detection is traditionally carried out by human experts. Human experts identify diseases visually but they faces some difficulties that may harm their efforts. In this context, detecting and classifying diseases in an exact and timely manner is of the great importance [1].

Advances in artificial intelligence researches now make it possible to make automatic plant disease detection from raw images [2]. Deep learning can be thought as a learning method on neural networks. One of the advantages of deep learning is that it can extract features from images automatically. The neural network learns how to extract features while training. CNN is a multi-layer feed-forward neural network and is the popular deep learning model.

In recent years, CNN models have been widely used in image classification problems. Lee at al. [3] introduce a hybrid model to extract contextual information of leaf features using CNN and Deconvolutional Networks (DN). Konstantinos at al. [4] performed several pre-trained CNN models on a large open leaves dataset. Their studies show that CNN is highly suitable for automatic plant disease identification. Durmus at al. [5] also used AlexNet and Squeeze pre-trained CNN models on tomato leaves from an open dataset to detect diseases. Atabay at al. [6] fine-tuned a pre-trained model and designed a new CNN model to perform tomato leaf disease identification. Their study indicates that custom CNN model gives better results than the pre-trained model. Setting a suitable CNN model is a challenging issue to produce higher accuracy values. Zhang at al. [7] proposed a three-channel CNN model based on RGB colors to detect vegetable leaf diseases.

Plant leaf images are complex with its background and the color information extracted from a single color component is limited. It causes the feature extraction method to give lesser accuracy results. Using different color components is promising instead of single one. In the proposed paper we developed a CNN model based on RGB components of the tomato leaf images on PlantVilliage dataset [8]. We preferred Learning Vector Quantization (LVQ) algorithm as classifier due to its topology and adaptive model.

The paper is organized as follows: section II gives details of CNN. Section III describes LVQ algorithm. Section IV provides the proposed method for plant leaf disease detection and classification. Section V evaluates experimental results. Finally, section VI concludes the paper.

II. Convolutional Neural Network

Deep learning is a class of machine learning algorithms that has sequential layers. Each layer uses the output of the preceding layer as input. The learning process can be unsupervised, supervised or semi-supervised. LeCun et al. define the deep learning as a representation learning method [9]. Representation learning algorithms makes optimizations to find the most convenient way to represent the data [5]. Deep learning does not have to divide the feature extraction and the classification because the model automatically extracts the features while training the model. It is used in many research

978-1-5386-7893-0/18/$31.00 ©2018 IEEE

areas such as image processing, image restoration, speech recognition, natural language processing and bioinformatics.

CNN is preferred as a deep learning method in this study. CNN, which can easily identify and classify objects with minimal pre-processing, is succesful in analyzing visual images and can easily separate the required features with its multi-layered structure. It consists of four main layers: convolutional layer, pooling layer, activation function layer and fully connected layer. Fig. 1 shows a general CNN architecture.

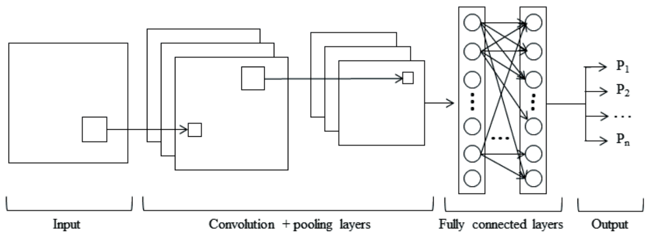

Fig. 1. A general CNN architecture.

A. Convolutional Layer

CNN takes its name from the convolution layer. In this layer, a series of mathematical operations are performed to extract the feature map of the input image [10]. The input image is reduced to a smaller size using a filter. The filter is shifted step by step starting from the upper left corner of the image. At each step, the values in the image are multiplied by the values of the filter and the result is summed. A new matrix with a smaller size is created from the input image. Fig. 2 shows the convolution operation in the convolution layer for a 5x5 input image and a 3x3 filter.

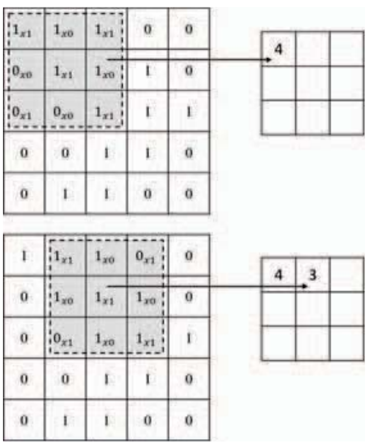

Fig. 2. Convolution operation for 5x5 input image and 3x3 filter.

B. Pooling Layer

The pooling layer is usually applied after the convolution layer. The size of the output matrix obtained from the convolution layer is reduced in this layer. Although filters of different sizes can be used in the pooling layer, generally 2x2 size filter is used. Several functions such as max pooling, average pooling and L2-norm pooling can be used in this layer. in this study, max pooling filter with stride 2 has been applied. Max pooling is done by selecting the largest value in the sub-windows and this value is transferred to in a new matrix. Fig. 3 shows an example pooling operation.

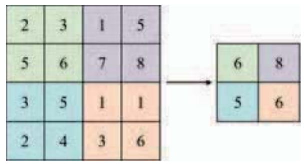

Fig. 3. Max pooling with 2x2 filters and stride 2.

C. Activation Layer

In artificial neural networks, the activation function provides a curvilinear relationship between the input and output layers. It affects the network performance. Non-linear learning of the network occurs through the activation function. Several activation functions, such as linear, sigmoid, hyperbolic tangent, exist, but the nonlinear ReLU (Rectified Linear Unit) activation function is usually used in CNN. In ReLU, values less than zero are changed to zero, while values greater than zero are unchanged by (1).

$$f(x) = \begin{cases} 0, & \text{if } x < 0. \\ x, & \text{otherwise.} \end{cases} \tag{1}$$

D. Fully Connected Layer

The last obtained matrix, after finishing the convolution, pooling and activation operations, is fed into the fully connected layer as input. Recognition and classification are performed in this layer. In this study, LVQ algorithm has been used for training the data classification. All layers are not fully connected in this study due to the structure of LVQ although many of studies in the literature use a fully connected structure.

III. Learning Vector Quantization

Learning Vector Quantization, proposed by Kohonen [11], is a neural network that combines competitive learning with supervised learning. It is a powerful and heuristic algorithm for solving classification problems. Due to its simple topology and adaptive model, LVQ has been widely used in many applications. It classifies the given data in a fixed number of classes [12]. It consists of three layers with input, Kohonen (competition) and output layer. Input layer neurons collect the values of the input variables. Each neuron in the output layer represents a class of input. The input and Kohonen layers are fully connected, while the Kohonen and output layers are partially connected. The learning occures in Kohonen layer. The classified results are passed to the linear output layer [13]. The LVQ architecture is shown in Fig. 4.

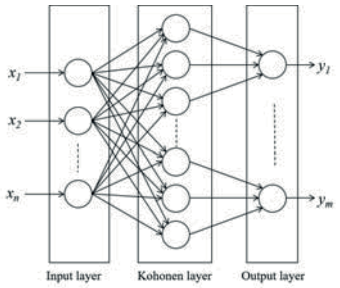

Fig. 4. The LVQ architecture.

In LVQ, the reference vectors consisting of weights are used to represent the classes for learning. The learning is based on the similarity between the input vector and reference vectors. Only one of the output takes the value 1 and the others take the value 0. The reference vector that receives the value 1 gives the class of the input vector. LVQ model works according to the "winner-takes-all" approach and only the weights of the winning reference vector which is closest to the input vector are updated at every iteration. The winning reference vector is found via calculating the Euclidean distance from the input vector to each of the reference vectors by (2);

$$d = \arg\min_{i} \{\|x - w_i\|\} \tag{2}$$

where x is input vector, w_i is i. reference vector. Reference vectors are updated by (3) if classification is correct, otherwise updated by (4):

$$w_i(t+1) = w_i(t) + \eta(t)(x - w_i(t)) \tag{3}$$

$$w_i(t+1) = w_i(t) - \eta(t)(x - w_i(t)) \tag{4}$$

where $\eta \in (0,1)$. η is the learning rate, and this rate is decreased monotonically with time. If the reference vector and input vector classes are matched, the reference vector is moved towards input vector. Otherwise, it is moved away from the input vector [14]. An operation of updating reference vectors repeats till classification rate is achieved or maximum number of epochs is reached.

IV. The Proposed Method for Plant Leaf Disease Detection and Classification

In this study, 400 training and 100 test tomato leaf images have been used from PlantVilliage dataset. The images in the selected dataset have been cropped to the size of 512x512. The intended leaf diseases to classify in this study are bacterial spot, late blight, septoria leaf spot and yellow leaf curl [15]. Five different classes have been used, four of them are for leaf diseases and one of them is for healthy leaves.

Bacterial Spot: Symptoms of bacterial spot begin as small, yellow-green lesions or as dark, water soaked, greasy-appearing lesions on leaves. Bacterial spot disease spreads very quickly and is very difficult to control. Therefore it is one of the most dangerous tomato disease. This disease can cause damage to the tomato plant and its marketability .

Late Blight: It is first seen as large brown spots with green-gray edges on old leaves. As the disease matures, the spots become darker. Eventually the disease infects the whole plant and causes the plant to be seriously damaged.

Septoria Leaf Spot: It first appears in the lower leaves of the plant. The symptoms are round, yellow, water-soaked spots that occur under the leaves. It causes small brown or black spots on the leaves. The size of the spots varies between 1.5 mm and 6.5 mm.

Yellow Leaf Curl: It causes the plant to become dwarfed and stunted. The leaves are rolled inwards and upwards. It usually causes the leaves to bend downwards. Leaves become stiff, thicker than normal and have a leathery skin texture. Young and diseased leaves become slightly yellowish.

Fig. 5 shows sample images of diseased leaves and a healthy leaf.

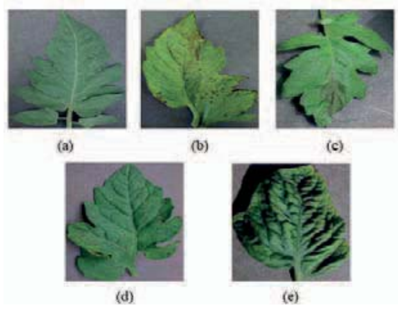

Fig. 5. (a) Healthy Leaf (b) Bacterial Spot (c) Late Blight (d) Septoria Leaf Spot (e) Yellow Leaf Curl.

Three different input matrices have been obtained for R, G and B channels to start convolution for every image in the dataset. Each input image matrix has been convoluted and reLU activation function has been implied four times, respectively. Then the max pooling operation has been implied to the output matrix three times. A 9x9 filter has been used in the first and second convolutions, and a 5x5 filter has been used in the third and fourth convolutions. After these implications, three different 3x3 matrices have been obtained for R, G and B channels separately. As a result of these operations applied to an RGB feature image, three different 3x3 matrices have obtained from R, G and B channels separately. These matrices are converted to a 27x1 vector in order to feed to neural network's input layer. The first 9 elements of this matrix represent the R channel, the second 9 elements represent the G channel, and the third 9 elements represent the B channel.

Total 500 feature vectors which obtained from original images have been used for training and testing operations. 400 of them were used for the training set, and 100 of them were used for the test set. In the LVQ algorithm, Kohonen layer of the network contains a total of 50 neurons which are 10 neurons for each class. The output layer contains 5 neurons to represent one neuron for each class. The maximum number of epochs has been selected as 300 in all experiments. The learning rate has been selected as 0.1.

Architecture of the proposed method is shown in Fig. 6.

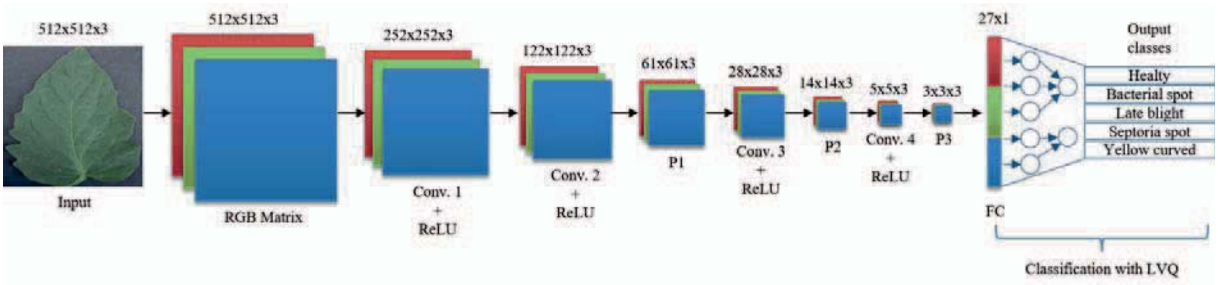

Fig. 6. Architecture of the proposed method.

TABLE I
CLASSIFICATION RESULTS AS CONFUSION MATRIX.

Leaf Disease	Healthy	Bacterial spot	Late blight	Septoria spot	Yellow curved	Accuracy
Healthy	18	0	0	0	2	90%
Bacterial spot	0	18	0	0	2	90%
Late blight	0	0	17	0	3	85%
Septoria spot	1	0	0	16	3	80%
Yellow curved	0	0	0	3	17	85%
Average						86%

V. EXPERIMENTAL RESULTS

To verify the performance of the proposed method, we have conducted a set of experiments on healthy and diseased tomato leaf image databases and have performed classiffication. One of the main challenges in disease detection and classification for this study is that the leaves with different diseases are very similar to each other. Therefore, this similarity can cause some leaves to be folded into to wrong classes.

The classiffication results are shown in Table I as confusion matrix. 20 images have been used for each class. As seen in the table, leaves ranging from 16 to 18 of 20 for each class have been correctly classified from these test data. Only a few leaves have been incorrectly classified for every classes, and it can be seen in which classes these wrong classifications have been folded in the table.

VI. CONCLUSION

In this paper, a tomato leaf diseases detection and classification method is presented based on Convolutional Neural Network with Learning Vector Quantization algorithm. The dataset consist of 500 tomato leaves images. Three different input matrices have been obtained for R, G and B channels to start convolution for every image in the dataset. Each input image matrix has been convoluted. reLU activation function and max pooling have been implied to the output matrix. Total 500 feature vectors which obtained from original images have been used for training and testing operations in the LVQ algorithm. The experiments have been carried out on healthy and diseased leaf images to perform classiffication. It is concluded that the proposed method effectively recognizes four different types of tomato leaf diseases. To improve recognition rate in classification process different filters or different size of convolutions can also be used.

REFERENCES

[1] H. Park, J. S. Eun and S. H. Kim, “Image-based disease diagnosing and predicting of the crops through the deep learning mechanism”, In Information and Communication Technology Convergence (ICTC), IEEE 2017 International Conference on, pp. 129-131, 2017.
[2] K. Elangovan and S. Nalini, “Plant disease classification using image segmentation and SVM techniques”, International Journal of Computational Intelligence Research, vol. 13(7), pp. 1821-1828, 2017.
[3] 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.
[4] K. P. Ferentinos, “Deep learning models for plant disease detection and diagnosis”, Computers and Electronics in Agriculture, vol. 145, pp. 311-318, 2018.
[5] H. Durmuş, E. O. Güneş and M. Kırcı, “Disease detection on the leaves of the tomato plants by using deep learning”, In Agro-Geoinformatics, IEEE 6th International Conference on, pp. 1-5, 2017.
[6] H. A. Atabay, “Deep residual learning for tomato plant leaf disease identification”, Journal of Theoretical & Applied Information Technology, vol. 95(24), 2017.
[7] S. Zhang, W. Huang and C. Zhang, “Tree-channel convolutional neural networks for vegetable leaf disease recognition”, Cognitive Systems Research, 2018.
[8] D. P. Hughes and M. Salathé, “An open access repository of images on plant health to enable the development of mobile disease diagnostics”, arXiv. 1511.08060, 2015.
[9] Y. LeCun, Y. Bengio and G. Hinton, “Deep Learning”, Nature, vol. 521, pp. 436-444, 2015.
[10] V. Tümen, Ö. F. Söylemez and B. Ergen, “Facial emotion recognition on a dataset using convolutional neural network”, 2017 International Artificial Intelligence and Data Processing Symposium (IDAP), November 2017.
[11] T. Kohonen, “Self-organization and associative memory”, third ed., Springer-Verlag, London, 1989.
[12] B. Mokbel, B. Paassen, F. M. Schleif and B. Hammer, “Metric learning for sequences in relational LVQ”, Neurocomputing, 169, 306-322, 2015.
[13] X. Li and Y. Zhang, “Digital image edge detection based on LVQ neural network”, 2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA), Hefei, China, pp. 1251-1255, 2016.
[14] J. S. Sánchez and A. I. Marqués, “An LVQ-based adaptive algorithm for learning from very small codebooks”, Neurocomputing, 69(7-9), 922-927, 2006.
[15] V. R. Preedy, ed., “Tomatoes and tomato products: nutritional, medicinal and therapeutic properties”, CRC Press, 2008.