Identification of Soybean Leaf Spot Diseases using Deep Convolutional Neural Networks

Jiangsheng Gui
Professor
Department of Information Science and Technology,
Zhejiang Sci-Tech University (ZSTU)
Hangzhou City, China.

Mor Mbaye
Student
Department of Information Science and Technology
Zhejiang Sci-Tech University (ZSTU)
Hangzhou City, China

***Abstract*— In this paper, we designed a Deep Convolutional Neural Network based on LeNet to perform soybean leaf spot disease recognition and classification using affected areas of disease spots. The affected areas of disease spots were segmented from the leaves images using the Unsupervised fuzzy clustering algorithm. The proposed Deep Convolutional Neural Network model achieved a testing accuracy of 89.84%, and poor per class recognition results in 1378 images misclassified, and 1271 images correct classified. TheVGG16 achieved the best performance reaching a 93.54% success rate, and better per class recognition results in 1245 images misclassified, and 1404 images correct classified.**

***Keywords*— Deep convolutional neural network, Unsupervised fuzzy clustering algorithm, image segmentation, soybean leaf spot diseases.**

I. INTRODUCTION

Most countries around the world have their economy that is dependent on agriculture. agriculture occupies an important place in the production and distribution of food to humans. (The growth of crops) the agriculture can not only bring necessities to people's daily life, but also improve soil fertility, maintaining a good soil ecosystem, control soil erosion, reduce natural disasters such as mudslides and sandstorms, and improve the environment in which humans depend. However, the growth and development of crops are closely related to the environment around them. and the environment becomes more and more polluted, may induce crop disease. Soybean is one of the most important agricultural products in the world [1]. Today, it is affected by several diseases that worry a lot of farmers, and the fight against crop diseases remains a major problem for them, to control these diseases, a large number of chemicals or fungicides are used on the citrus crop, which results in both economic loss and environmental pollution [2]. Now the new technologies based on artificial intelligence can develop precision agriculture, improve crops, manage and limit the misuse of chemicals in the beds [3,4]. Detection and classification of plant diseases are important tasks to increase plant productivity and economic growth [5,6,7]. computer vision, machine learning, and deep learning algorithms make it possible to develop tools for control and analysis of plant diseases [2,8,9].

II. RELATED WORK

Today computer vision, machine learning, and deep learning are used in different fields, with the power of their classification algorithms and image analysis to revolutionize the world of agriculture. Different approaches have been developing for crop disease classification and detection:

Machine learning methods, such as artificial neural networks (ANNs), Decision Trees, K-means, k nearest neighbors, and Support Vector Machines (SVMs) have been applied in agricultural research [10,11,12,13].

In [14] Shinde PG, Shinde AK, Shinde AA, Borate SP (2017) used image processing technique (k-means clustering) and raspberry PI to develop a system for automated crop disease which made use of email alerting and SMS functionalities to predict the disease and pesticide name.

In [15] Baghel J. (2016) proposed k-means clustering segmentation technique, their proposed algorithm analyses the area of the leaf to separate the infected part and uninfected part of the leaf.

Recently convolutional neural networks(CNN) have been used for object recognition and image classification:

In [16] Serawork Wallelign, Mihai Polceanu, C´edric Buche (2017) based on the LeNet architecture was defined a CNN model to identify and classify 12,673 leaf images of soybean crop with 3 classes symptom images (Septorial leaf blight, Frogeye leaf spot, Downy Mildew), and the proposed algorithm achieved a good result with success rate of 99.32% in the classification of four different class soybean diseases.

In [17] Ma J, Du K, Zheng F, Zhang L, Gong Z, Sun Z (2018) designed a new CNN model similar to Lenet5 and adopted the same image processing technique in their previous work [18]. The architecture of the model was composed of four modules, the first module is a convolutional layer that had 20 filters with a size of 5× 5 followed by ReLu activation and Max-pooling layers that has size of 2 x 2 and stride of 2, second module a convolutional layer that had 100 filters with a size of 3× 3, and a Max-pooling Layer with the filter that had a size of 2×2 and a stride of 2, the third module consisted of a Convolutional Layer that had 1000 filters with a size of 3× 3, the last module consisted of a Fully Connected Layer with 1500 neurons. The algorithm was applied to deal with four different cucumber diseases, when the algorithm was applied on the segmented symptom image the accuracy was 93.4%, and when the algorithm was implemented to identify symptom images under the influences of illumination the results were 98.1% of accuracy. When they use data augmentation methods to enlarge the datasets formed by the segmented symptom images, the model achieved 93.47% of accuracy.

In [7] Ferentinos KP (2108) used the existing deep CNN architectures AlexNet [19], AlexNetOWTBn [20], GoogLeNet [21], Overfeat [22], and VGG [23] to classify plant diseases. Using a public dataset of 87,848 images of diseased and healthy plant leaves collected under controlled conditions, the CNN was trained to identify 25 different plants in a set of 58 distinct classes of plant, diseases. The best performance reaching a 99.53% success rate in identifying the corresponding plant, disease combination healthy plant.

The main objectives of this study were to recognize and classify the soybean leaf spot disease using the affected areas of disease spots based on the Deep Convolutional Neural Network.

III. MATERIAL AND METHODS

A. Datasets

The datasets used for this work were downloaded from different databases on the internet (https://www.forestryimages.org https://plantvillage.org, http://www.image-net.org/challenges/LSVRC/2012/). The dataset includes 13243 leaf images of soybean crop with 5 classes of soybean leaf spot disease (Alternaria Leaf Spot, Phyllosticta Leaf Spot, Target Leaf Spot, Frogeye Leaf Spot, Bacterial Blight). TABLE 1 summarizes the number of images for each type of soybean leaf spot disease.

The symptom images were segmented using the Unsupervised fuzzy clustering algorithm which combines the advantages of fuzzy mean algorithm and unsupervised optimal clustering algorithm. It is based on the fuzzy mean algorithm. By gradually increasing the number of clusters and evaluating according to the effectiveness, it can find the best without supervision. The number of clusters. Compared with the fuzzy mean algorithm, the unsupervised fuzzy clustering algorithm improves the distance function, so that the cluster is not interfered with by the shape of the class so that the number of clusters can be accurately found and the clustering effect can be achieved more accurately. The symptom images were segmented following these steps:

1) Select the initial cluster center, and set the contrast factor, the maximum allowable error and the maximum number of clusters.
2) An initial clustering model can be obtained by clustering by a fuzzy mean algorithm. Use Euclidean distance in the distance function selection.
3) Again using the fuzzy mean algorithm for clustering, unlike step (2), the distance function is changed to an exponential distance function, as follows:
4) Calculate the effectiveness of clustering to measure scale parameters, there are some parameters below Fuzzy oversize standard:
5) If the current cluster is smaller than the predetermined maximum number of clusters, the number of clusters plus one is recalculated in step (2), otherwise, the calculation is stopped and the validity criterion is selected to select the cluster number with the largest average separation density as the best clustering.

Data that were of different sizes, were all being resized to 128 x 128 pixel. the dataset was divided into two sets, 80% for the training and 20% for testing. Shown in TABLE 1.

TABLE 1: a training set and test set used for the classification

No.	**Diseases name**	**Number of images**	**Training set**	**testing set**
1	Alternaria Leaf Spot	1825	1460	365
2	Phyllosticta Leaf Spot	3350	2680	670
3	Target Leaf Spot	2921	2338	583
4	Frogeye leaf spot	2945	2356	589
5	Bacterial blight	2202	1760	442
Total		13243	10594	2649

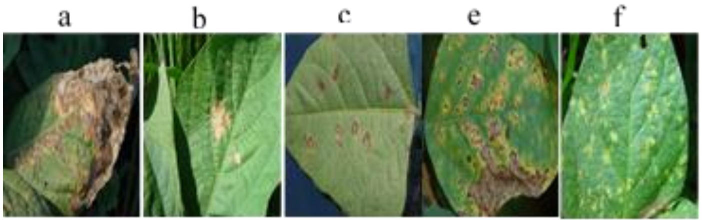


Fig. 1: Sample of soybean leaf spot diseases taken in the real cultivation condition in the field. a: Alternaria Leaf Spot, b: Phyllosticta Leaf Spot, c: Target Leaf Spot, d: Frogeye Leaf Spot, e: Bacterial Blight.

B. Deep Convolutional Neural Network model (Deep CNN)

The proposed CNN model used this work consist of 5 convolutional layers (Conv), 2 Max-Pooling layers, and 2 fully connected layers. 32 x 32 input mages size was selected for the classification. The architecture of this trained CNN model is shown in Fig.2. The first two convolutional layers (Conv1, and Conv2) are composed of 32 kernels of size 3 x 3, followed by the ReLu activation function forces the neurons to return positive values, and Max-poling with the filter that had a size of 2 x 2 and stride 3. The third convolutional layer (Conn3) consists of 64 kernels of size 4 x 4, followed by the Relu activation function, and the Max pooling layer had filter size 2 x 2 and stride 2. And the last two convolutional layers (Conv4, and Conv5) are composed of 128 filters with a size 3 x 3 followed also by Relu activation function, and the Max pooling layer had filter size 2 x 2 and stride 2. The two fully connected layers, each has 1024 neuron. ReLu activation and Dropout were applied after each fully connected layer. The probability of randomly drop a unit is 25% for the first one and 50% for the second Dropout. The softmax output classification was implemented to calculate the probability distribution of the 5 classes symptom images.

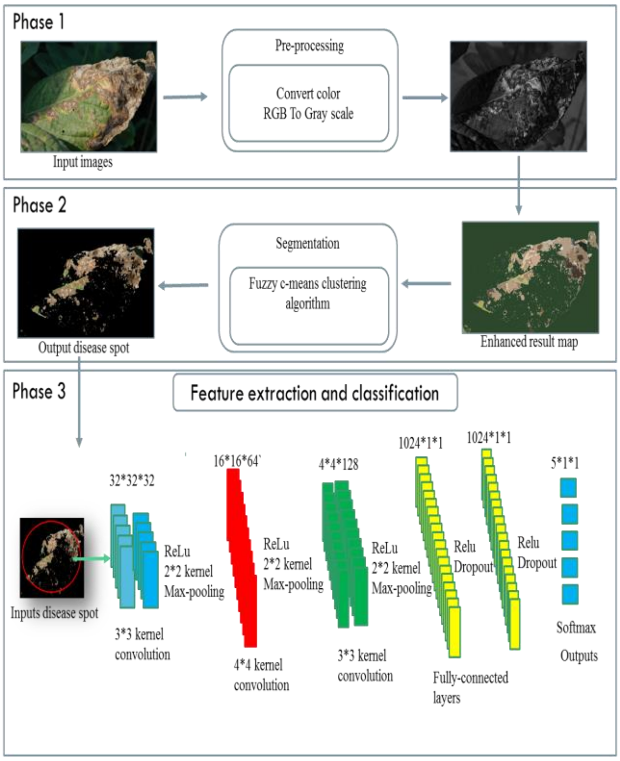


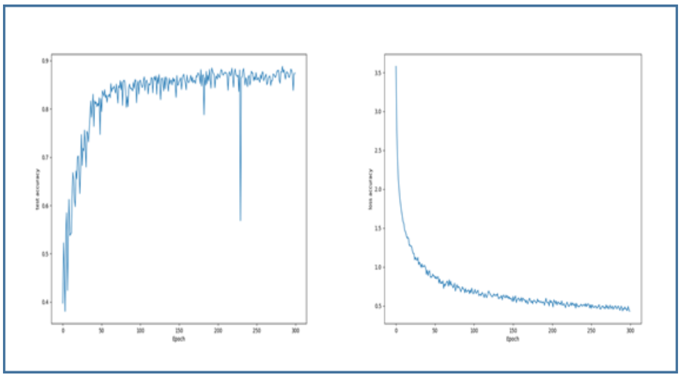
Fig. 2: Image segmentation process and Deep CNN architecture

IV. RESULTS AND DISCUSSION

C. Performance model

As mentioned earlier, during the training stage, the dataset was divided into two sets, 80% for the training, 20% for testing as shown in TABLE 1. The proposed CNN model was loaded on python programming language using Keras and TensorFlow's deep learning library. To enlarge the datasets and overcome overfitting some simple and efficient technique was implemented such as data augmentation and Drop out. These algorithms are trained with a batch size of 32 for 1000 epochs with a momentum of 0.9, weight decay of 0.0005, and a learning rate of 0.001. Training algorithms were implemented on the GPU of a GeForce GTX1070 card, using the CUDA parallel programming platform, in a Linux environment (Ubuntu 16.04 LTS 64-bit operating system).

As we can see from TABLE 2, the designed CNN model achieved good recognition results, with a success rate of 89.84%, and an average loss of 0.20%. Fig. 3 illustrates the testing accuracy and loss accuracy of the Deep CNN model on the testing dataset.



(a) (b)

Fig. 3 Testing accuracy (a), and loss accuracy (b), of the Deep CNN model on the testing dataset.

TABLE 2: the percentage of success of the proposed model with the original image

Models	Val accuracy(%)	Loss(%)	epochs	times
DCNN models	89.84	0.20	250	21:54:01

D. Recognition performance

In this work, I use the three metrics of the confusion matrix: Precision, Recall, F-score, which can be respectively calculated from equations 1, 2 and 3 to measure the per class recognition performance of the DCNN algorithm.

$$Precision = \frac{\#TruePositive}{\#TruePositive + \#FalsePositive} \quad (1)$$

$$Recall = \frac{\#TruePositive}{\#TruePositive + \#FalseNegative} \quad (2)$$

$$F1_{-Score} = \frac{2 * Precision * Recall}{Precision + Recall} \quad (3)$$

The per-class recognition performance metrics of the DCNN model are summarized in TABLE 3.

In the performance per class shown in Fig. 4, on the Alternaria Leaf Spot class 286 symptoms images were correctly classified, 79 images misclassified, with a probability rate of 85.32%. On Phyllosticta Leaf Spot class 498 images were correctly classified, 172 images misclassified with a probability rate of 87.13%. On the Target Leaf Spot, 214 images were correctly classified, 369 misclassified with a probability rate of 77.08%. On the Frogeye leaf spot 158 images correctly classified, 431 images misclassified with a probability rate of 75.03%. On the Bacterial blight class, 115 images correct classified, 327 images misclassified with a probability rate of 78.45%.

TABLE 3 Per class recognition performance metrics of the DCNN

	Alternaria Leaf Spot	Phyllosticta Leaf Spot	Target Leaf Spot	Frogeye leaf spot	Bacterial blight	Precision	Recall	F1-Score
Alternaria Leaf Spot	286	10	22	15	32	89.56	91.45	89.03
Phyllosticta Leaf Spot	125	498	3	27	17	82.45	85.12	85.12
Target Leaf Spot	157	98	214	76	38	86.43	78.45	86.23
Frogeye leaf spot	127	37	29	158	238	75.03	83.23	77.45
Bacterial blight	98	45	72	112	115	78.45	81.05	75.00

As we can from those results, the implemented CNN algorithms didn't achieve a good performance per class recognition, the number of symptoms images misclassified (1378 images, probability rate of 52%) was larger than the number of symptom images correct classified (1271 images, probability rate of 48%), see in Fig. 5. This poor recognition performance can be explained by the reason that the spot diseases were segmented from input images. during the image recognition process, Deep CNN can logically analyze these constructs, first by simplifying the image and extracting the most important information, then organizing by data through feature extraction and classification.

We hypothesized that these results may be due to the quality of the images, and the characteristics such as the shape, the texture, and the margin extract from the input images [24].

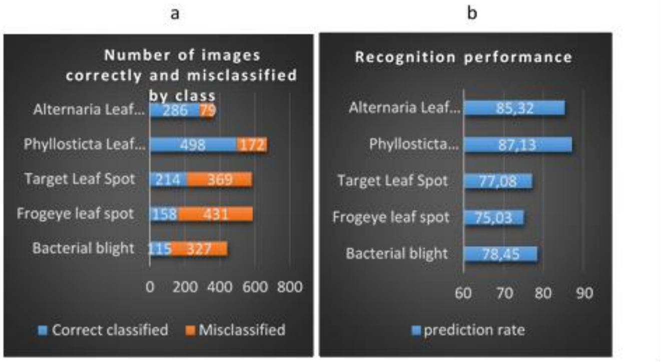


Fig. 4: Per class recognition results with Deep CNN model

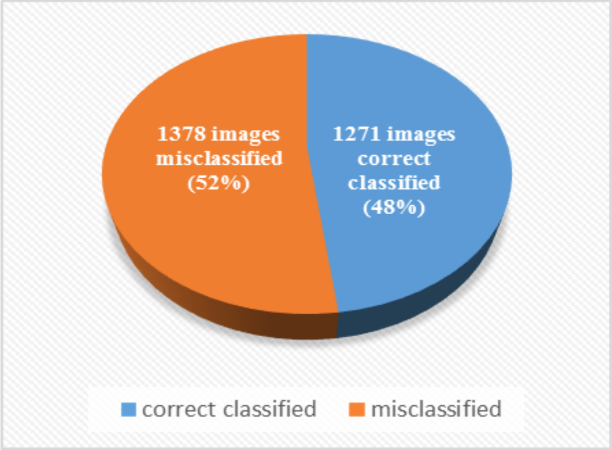


Fig. 5: Total number of images correct classified and misclassified with Deep CNN model

E. *performance evaluation*

The VGG 16 architecture introduced by [23] and SVM introduced by [25] was adopted In this work to evaluate the performance of the proposed CNN model to recognize four classes of soybean leaf spot diseases. The model evaluated with the same, and an equal number of images for each class.
It can see from the TABLE 4 the VGG16 achieved a testing accuracy of 93.45 and per class recognition performance of 1404 (prediction rate of 53%), symptoms images correct classified and 1245 images misclassified (prediction rate of 47%). The SVM achieved a testing accuracy of 83.23% and per class recognition performance 1106 (prediction rate of 42%) symptom images correct classified, 1543 misclassified (prediction rate of 58%).
The results presented in TABLE 4 indicated that VGG16 achieved better test accuracy and better per class recognition performances than the Deep CNN model. The proposed Deep CNN model achieved better test accuracy and better per class recognition performances than the SVM.

TABLE 4 Performance of different implemented models for the soybean leaf spot disease recognition

	Test accuracy	Loss accuracy	epochs	Training time	Correctly classified	Misclassified
VGG16	93.45	0.29	123	22:15:32	1404 images (53%)	1245 images (47%)
Deep CNN Model	89.84	0.37	176	21:54:01	1271 images (48%)	1378 images (52%)
SVM	83.23	0.48	150	1days/16:32:09	1106 images (42%)	1543 images (58%)

As shown on the per-class recognition results in Fig .4a, Fig. 6a, and Fig. 7a, all the implemented models (Deep CNN model, VGG16, SVM) outperformed a similar per class recognition performance on the Target Leaf Spot class, Frogeye Leaf Spot class, and Bacterial Blight class, shown in Fig.10. The similar prediction results of these soybean leaf spot disease classes may be caused by the reasons that the pattern of their spot diseases on the leaves area are identical. Their spot diseases are characterized by small, pale green spots or streaks which soon appear water-soaked, circular to angular in shape, shown in Fig.8. The implemented algorithms have difficulty predicting the correct class using the spot diseases segmented. These results verify the hypothesis described in Section 4.2. we conclude, however, the quality, the characteristics of the input images such as the shape, the texture, and the margin are an influence of factor to the Deep CNN. so for image recognition, it is better to implement the inputs images without extracting any feature any needed information from the input images and. Because the Depp CNN needs to learn more information during the training stage about the input images and Deep CNN have the ability for feature extraction. In deep learning, each level of layer learns to transform its input data into a slightly more abstract and composite representation. Each layer has a specific task for image processing, for example:

- Convolution layer (CONV) that processes data from a receiver field.
- The pooling layer (POOL), which allows compressing information by reducing the size of the intermediate image (often by sub-sampling).
- The correction layer (Relu), often called by abuse 'Relu' concerning the function activation (linear grinding unit).
- The "fully connected" (FC) layer, which is a perceptron type layer.

The future work will be to implement Deep CNN for leaf disease recognition and classification under the real cultivation condition in the field.

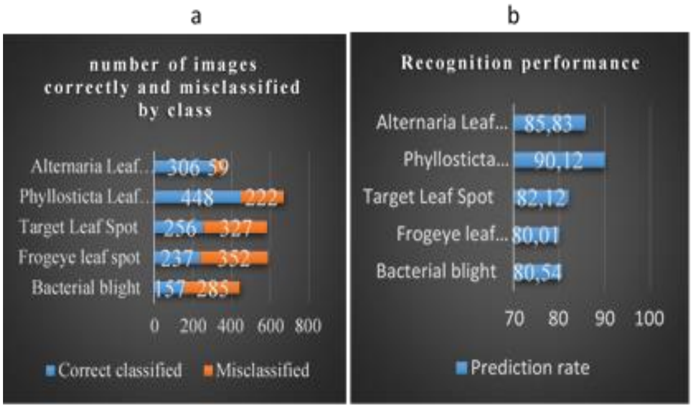


Fig. 6 Per class recognition results with VGG16

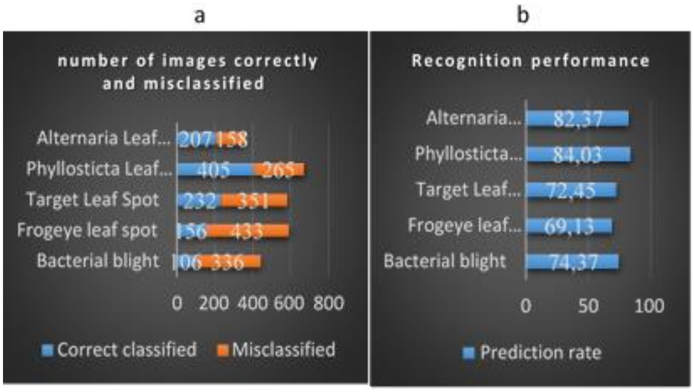


Fig. 7 Per class recognition results with SVM

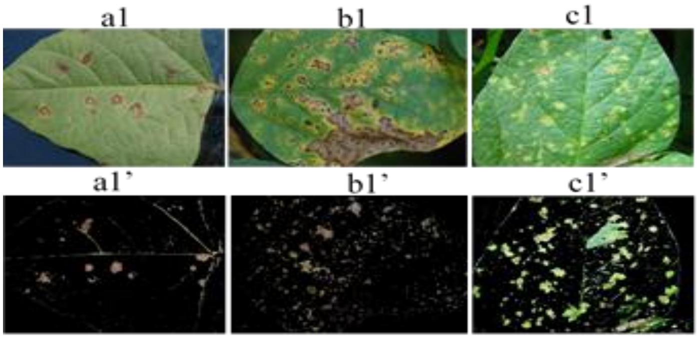


Fig. 8. a1: Target Leaf Spot, b1: Frogeye Leaf Spot, c1: Bacterial Blight with segmented results a1', b1', c1'.

V. CONCLUSION

In this paper, we presented the Deep convolutional neural networks. The results show that Deep CNN has difficulty to recognize the leaf spot disease and predicting the correct class using the spot diseases segmented. Comparing the results, All the implemented models (Deep CNN model, VGG16, SVM) outperformed a similar per class recognition performance. Moreover, it is better to analyze the data obtained without the extraction of any feature for the input images using DCNN, because with these convolutional layers have the ability to recognize the important features and non-important features for the input images.

ACKNOWLEDGMENT

This work was supported by the National Science Foundation (61105035, 61502430).

REFERENCES

[1] Pagano MC, Miransari M. production worldwide. Elsevier Inc.; 2016. DOI:10.1016/B978-0-12-801536-0/00001-3.

[2] Ali H, Lali MI, Nawaz MZ, Sharif M, Saleem BA. Symptom-based automated detection of citrus diseases using the color histogram and textural descriptors. Computers and Electronics in Agriculture 2017;138:92–104. doi:10.1016/j.compag.2017.04.008.

[3] Milioto A, Mar C V. Real-time Semantic Segmentation of Crop and Weed for Precision Agriculture Robots Leveraging Background Knowledge in CNN's 2018.

[4] Paszke A, Chaurasia A, Kim S, Culurciello E. EN ET : A DEEP NEURAL NETWORK ARCHITECTURE FOR REAL -TIME S SEMANTIC SEGMENTATION 2017:1–10.

[5] Lecun Y, Bengio Y, Hinton G, Jordan MI, Berkeley UC, Hinton G. Deep Learning Authors ’ Relationships 2015;444:436–44. doi:10.1038/nature14539.

[6] Gutte VS, Gitte MA. A Survey on Recognition of Plant Disease with Help of Algorithm 2016;6:7100–2. doi:10.4010/2016.1691.

[7] Ferentinos KP. Deep learning models for plant disease detection and diagnosis. Computers and Electronics in Agriculture 2018;145:311–8. doi:10.1016/j.compag.2018.01.009.

[8] Lauer F, Guermeur Y, Lauer F, Guermeur Y, Vector MS, Package M, et al. MSVMpack : a Multi-Class Support Vector Machine Package To cite this version : HAL Id : Hal-00605009 MSVMpack : A Multi-Class Support Vector Machine Package 2011.

[9] Chapelle O. Support Vector Machines et Classification d ’ Images Support Vector Machines for Image Classification 1998.

[10] Machines MSV. !()+, -./01 23456 1998:0–9.

[11] Chao C, Horng M. The Construction of Support Vector Machine Classifier Using the Firefly Algorithm 2015;2015. doi:10.1155/2015/212719.

[12] Zammit O, Descombes X, Zerubia J, Inria-is A, K-moyennes A. Apprentissage non supervis ´ e des SVM par un algorithme des K-moyennes entropique pour la d ´ etection de zones brˆ ul ´ ees 2007:11–4.

[13] Zhang C, Pan X, Li H, Gardiner A, Sargent I. A hybrid MLP-CNN classifier for very fine resolution remotely sensed image classification n.d.:1–33.

[14] Shinde PG, Shinde AK, Shinde AA, Borate SP. Plant Disease Detection Using Raspberry PI By K-means Clustering Algorithm 2017:92–5.

[15] Baghel J. Disease Detection in Soya Bean using K-Means Clustering Segmentation Technique 2016;145:15–8.

[16] Wallelign S. Soybean Plant Disease Identification Using Convolutional Neural Network 2017:146–51.

[17] Ma J, Du K, Zheng F, Zhang L, Gong Z, Sun Z. Original papers A recognition method for cucumber diseases using leaf symptom images based on deep convolutional neural network. Computers and Electronics in Agriculture 2018;154:18–24. doi:10.1016/j.compag.2018.08.048.

[18] Ma J, Du K, Zhang L, Zheng F, Chu J, Sun Z. A segmentation method for greenhouse vegetable foliar disease spots images using color information and region growing. Computers and Electronics in Agriculture 2017;142:110–7. doi:10.1016/j.compag.2017.08.023.

[19] Krizhevsky A, Hinton GE. ImageNet Classification with Deep Convolutional Neural Networks n.d.:1–9.

[20] Krizhevsky A. One weird trick for parallelizing convolutional neural networks 2014.

[21] Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, et al. Going Deeper with Convolutions 2014.

[22] Sermanet P, Eigen D. OverFeat : Integrated Recognition , Localization and Detection using Convolutional Networks arXiv : 1312 . 6229v4 [cs . CV] 24 Feb 2014 n.d.

[23] Arge FORL, Mage CI. V d c n l -s i r 2015:1–14.

[24] Dodge S, Karam L. Understanding How Image Quality Affects Deep Neural Networks n.d.

[25] Cortes C, Vapnik V. Support-Vector Networks 1995;297:273–97.