Atlantis Highlights in Computer Sciences, volume 4
Proceedings of the 3rd International Conference on Integrated Intelligent Computing Communication & Security (ICIIC 2021)

State-of-Art Deep Learning Based Tomato Leaf Disease Detection

Asha Gowda Karegowda [1], Raksha Jain [2], Devika G[3,*]

[1,2] Department of MCA, Siddaganga Institute of Technology, Tumkur, Karnataka, India
[3] Department of CSE, Govt. Engineering College, KR Pet, Karnataka, India
[*]Corresponding author. Email: sgdevika@gmail.com

ABSTRACT
In India, the tomato plant is a popular staple food with high commercial value and considerable production capacity; however, the quality and quantity of the tomato harvest decreases due to a variety of diseases and henceforth leads to great financial loss for farmers. With lack of agricultural professions to assist the farmers, a deep learning (DL) based user friendly, just-in-time mobile is proposed for the detection of crop diseases for assisting farmers to know about the type of tomato disease and the remedy for the same. Two DL based methods: YOLO and Faster RNN have been used for detection; followed by classification using SVM and Random forest tree. YOLO and Random forest tree resulted in accuracy in the range of 90% to 95%. The developed app provides option to the farmer to operate in English as well as in local language Kannada of Karnataka state of India.

Keywords: *Faster RNN, Random Forest Tree, SVM, Tomato Leaf disease detection, YOLO.*

1. INTRODUCTION

India is highly populated country whose Gross Domestic Product (GDP) is mainly reliable on agriculture. Tomatoes are the most common veggie used in Indian day to day cookery. Vitamins A, C, and E are the 3 maximum effective antioxidants in Tomatoes. Tomatoes are excessive in diet E, diet C, and beta-carotene. They're additionally excessive in potassium, that is critical for exact health. Mineral of significance Tomatoes is grown in India, covering roughly over 3,50,000 hectares.

Plants affected by Tomato diseases account for 10-30% of total crop losses and hence huge financial loss to the farmers. Manual monitoring of plant diseases is a difficult task because of its complexity, lack of professionals and time consuming process. Currently, farmers request the agricultural professions to visit their farms to check for the diseases or they need to take sample of tomato leaf to the agriculture office. This is not only time consuming, but also incurs financial loss, if actions are not taken in time. Hence there is need for just-in-time, farmer friendly disease detection mobile app. The proposed work is carried out using two Deep Learning techniques: YOLO (version 1 and version 3) and Faster RNN for detection of 9 varieties of tomato leaf disease in tomato leaf followed by classification using SVM and Random forest tree. The proposed app also provides information about the area affected by the disease and the type of and amount of pesticide to be used to overcome the issue. In addition, the proposed app provides farmer to select language for using the mobile app. currently the proposed work provides two languages options: English and local Kannada language of Karnataka state, India. This project supports both web platforms and Android applications. For the proposed system, Python and Android Studio as well as the Xampp software consisting of two modules are used.

The paper is presented as follows. Section 2 briefs about the related work. Brief description of 2 deep learning techniques: YOLO and Faster RNN adopted for the proposed work followed by proposed algorithm is discussed in Section of 3. Results and discussion are provided in Section 4 and 5 respectively followed by conclusion in Section 6.

2. RELATED WORK

Mim *et al.* [1] used CNN to detect 5 popular tomato leaf diseases and obtained validation accuracy of 96.55 % after 30 epochs. Lu *et al.* [2] adopted CNN approach to classify 10 rice diseases and obtained an average accuracy of 95.48%. In [3] authors have

Copyright © 2021 The Authors. Published by Atlantis Press International B.V.
This is an open access article distributed under the CC BY-NC 4.0 license -http://creativecommons.org/licenses/by-nc/4.0/.

employed Raspberry Pi with image processing for tomato plant disease Detection using exploitation K means, CNN and SVM classifiers using 4 factors: soil, temperature, and status and actinic ray rays. Ma *et al.* [4] obtained accuracy of 93.4$ for four types of cucumber diseases using CNN.

ResNet-101 took more time for training and detection but achieved higher detection rate compared whereas MobileNet has fast detection but low accuracy of tomato fruit disease [5]. An accuracy of 96.6% is obtained for identification of 39 kind of leaves using nine-layer CNN [6]. In [7], authors have carried out extensive survey on plant disease detection using various DL models and about performance metrics.

Wilt virus in tomato is detected using variation of Generative Adversarial Networks (GAN) with Outlier removal and Auxiliary classifier with hyperspectral Imaging to get an accuracy of 96.25% [8]. SVM with RFB kernel was used for detection of disease in capsicum and tomato plant leaf using h hyperspectral imaging and obtained accuracy more than 90 % in VNIR and SWIR spectrum ranges [9]. In [10] authors have proposed modified CNN (3 convolution→ 3 max pool → 2 fully connected layers) with varying number of filters and achieved average accuracy of 91.2% for tomato leaf disease detection compared 3 pre-trained model(VGG16, InceptionV3 and MobileNet)

Nachtigall *et al.* [11] used AlexNet for detecting and classifying nutritive deficits on apple tree and damage on apple trees to obtain accuracy of 97.3% and noted that MLP obtained 77.3%. DeChant *et al.* [12] classified northern leaf blight lesions on maize plant images to and obtained accuracy of 96.7 using ensemble of CNN.

Lu *et al.* [13] identified ten paddy diseases using AlexNet to achieve accuracy of 95.48%. In [14], 9 types of tomato disease are detected using five pre-trained models: AlexNet, GoogleNet, Inception V3, ResNet18, and ResNet 50 and found GoogleNet showed superior performance with 99.72% accuracy.

Performance of pretrained CNNs: Densenet 120, (Residual Network) ResNet 101, ResNet 50, ReseNet 30, ResNet 18, SqueezeNet and Vgg.net (Vgg-19 and Vgg-16) is carried out for tomato leaf disease identification and DenseNet 120 CNN obtained 99.81% accuracy [15]. A 3D CNN model is developed for identifying charcoal rot disease in soyaben using hyperspectral images and got accuracy of 95.73% [16]. Darknet-53 model is proposed for disease location and detection in Tomato, Corn and potato plant and obtained better results compared Mobile Net, Dark Net-19, ResNet-101and proposed model out PERFORMS in location and detection of plant diseases [17].

3. METHODOLOGY

3.1. Deep Learning

Machine learning is a type of artificial intelligence where the system learns from its experiences and improves the experience without the need for programming. It focuses on the growth of the computer program so that the data obtained can be used for self-education. The major advantage of deep learning compared to traditional machine learning is that, the former has capability of feature extraction, provide better accuracy and can be adopted for text, audio and image/video data, but needs large volume of input data and are demand high processing powered systems. YOLO, and RNN and it various versions are commonly used DL models for object detection [17-20]. These are briefed for the sake of completeness.

3.2. YOLO Algorithm

The YOLO technology, which stands for “You Only Look Once”, uses convolutional neural networks (CNN) to recognize objects in real time as shown in figure1. In order to recognize objects, the approach, as the name suggests, only requires a single forward propagation through a neural network, so that the entire image is captured in a single pass of the algorithm. Convolution networks are used to identify different class probabilities and bounding boxes at the same time. There are several variations of this method, the most common of which are the YOLO and YOLOv3. The advantages of YOLO are: The speed of detection is high and can detect objects in real-time; It is a predictive technique that provides accurate results with fewer background errors [20-23].

YOLO’s Operation (figure2): It employs three methods to accomplish its goal:

- Residual blocks: The image has been divided into several grids. The dimensions of each grid are S x S.
- Bounding box regression: An outline is the tip of an object in an image. The following properties apply to each bounding box in the picture: 1. Length (BW) 2. Dimensions (BH) 3. Educational level (c)
- Intersection Over Union (IOU): Intersection over Union is a statistic for determining how accurate an item detector is on a given dataset.

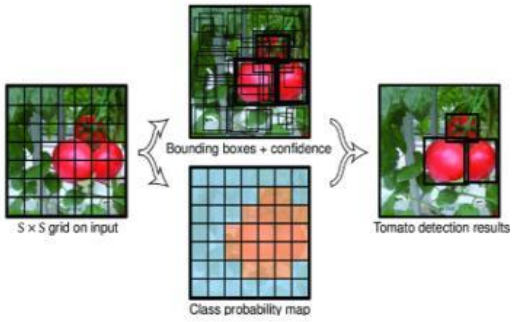


Figure 1 Working of Yolo for object detecton

3.2.1. YOLO v1 Architecture

There is various version of YOLO. YOLO [figure 2] is considered as a CNN which is extremely fast during inference. Instead of treating object detection as a classification problem, it approaches it a regression problem. It learns globally rather than locally by looking at the full input picture. All components for object recognition are combined in a single neural network. Each bounding box is predicted based on the properties of the entire image. It predicts all bounding boxes for an image in all classes at the same time, since you only look once at the input image. YOLO has been trained to recognize ten different disease classes. The input image of the network is segmented into a grid with the dimensions S x S. Each cell in the grid must determine whether or not it contains the center of an object from one of the ten disease classifications. Bounding boxes are a method of locating things in a photo or video. [24-26]

A bounding box consists of four dimensions: the x and y coordinates of the object center and the height and width of the box relative to the origin of the grid cell. The top left corner of the grid cell is commonly referred to as the cell's origin. There are five parameters in each bounding box: y = [pc, bx, by, bh, bw] where

i. pc = Po x IOU: The bounding box confidence value The bounding box confidence value indicates how confidently the model predicts the object and how accurately it assesses the bounding box one has created. Union (IOU)between the predicted box and the fundamental truth to get 'PC'.

ii. bx and by are Object Center Coordinates. The x and y coordinates of the center point of the element recognized by the grid cell are measured from the top left corner of the grid cell, the origin.

iii. bh and bw are bounding box height and width The height and width of the grid cell's estimated bounding box, measured from the cell's origin.

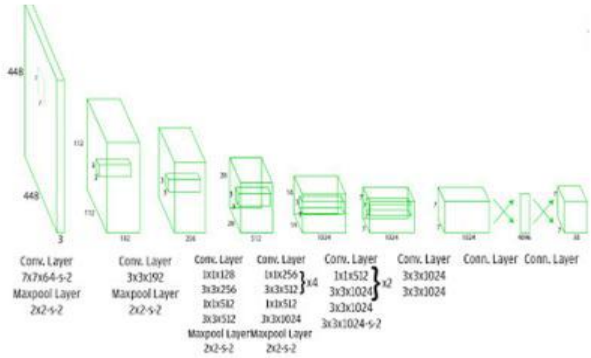


Figure 2 Yolo version 1 Architecture

Each cell in the grid has class 'C' probabilities which are simply the conditional probabilities of the type of object present in the bounding box called Pr (class object), plus the bounding box and the confidence score for each box. Therefore, to recognize these "C" classes, a grid cell can generate class "C" probabilities, but only one set of class probabilities is predicted per cell. That is, in YOLO v1 each cell in the grid can only anticipate one thing, for example a cat, a dog or a car. One cannot predict the presence of numerous elements such for example, as a cat and a dog in the same grid cell. This is a major bug in YOLO v1. Cannot identify and locate more than one object. Per grid cell as a result, the output of the S x S grid has the following dimension: S x S x (B x 5 + C).

The tomato leaf detection dataset was used to test YOLO version1, which is constructed as a convolutional neural. There are three in total Folding layers followed by two fully connected layers. The dataset was pretrained with three convolutional layers, three maximal grouping layers, and two fully connected layers. The classification pre-training is carried out on a dataset with a resolution of 448 x 448 pixels. The 1x1 reducing layers and 3x3 folding layers make up the layers. To train the network to recognize objects, the last three layers of convolution are added, followed by two fully connected layers.

Object detection necessitates greater granularity; therefore, the dataset's resolution is increased to 8 × 8. The class probabilities and bounding boxes are predicted in the final layer. The final convolutional layer utilizes linear activation, while the others use leaky ReLU activation. The input is an image, and the output is the object's class classification within the bounding box. In addition, the proposed work is compared with YOLO version 3. The major differences between the basic YOLO version (v1) and the version 3 are briefed in the Table 1.

Table 1. Difference between Yolo v1 and Yolo v3

Yolo v1	**Yolo v3**
Yolo v1 consists of 24	YOLO v3 consists 53

convolutional layers .	convolutional layers. And 53 layers for detection are added making it as 106 layers.
1x1 reduction layers and 3x3 convolutional layers comprise up the layers.	Implementation of 1 x 1 detection kernels to feature maps of three different sizes at three distinct locations in the network is used to identify the network.
The input is a square-shaped image with a resolution of 448 x 448 pixels.	The model anticipates color images with a square form of 416 x 416 pixels as inputs.

3.3. Faster r-cnn Architecture

The working of object detection algorithm Faster R-CNN is briefed as follows. When an image is passed to Regional Proposals Network (RPN) (figure 3), it should get a number of bounding boxes back. These bounding boxes, on the other hand, are used by the object recognition network to classify the object. The region suggestions from RCNN are taken directly from feature maps in order to be Faster RCNN. To address this problem, the RPN was created. [26-28]

Bounding boxes are defined as anchors. Three scales (128x 128, 256 x 256 and 512 x 512) and three ratios (1: 1, 1:2 and 2: 1) are the standard anchors, so it has a total of 9 anchors. The scale size, proportion and number of anchors are determined by the application. For example, the scale for recognizing faces in photos usually does not have to be very large.

The output of this layer goes through two 1 * 1 convolution layers, the classification layer and the regression layer. The classification layer has 2 * N (W * H * (2 * k) output parameters, while the regression layer has 4 * N (W * H* (4 * k) output parameters (denotes the coordinates of the bounding boxes) (denotes) object probability or not).

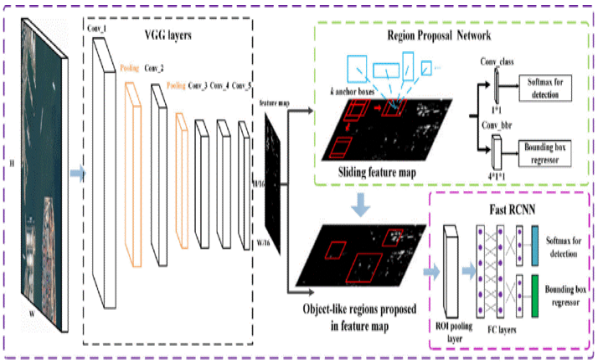


Figure 3 Faster R-CNN Architecture

The first layer predicts the objects in the proposed region using a Softmax layer with N + 1 output parameters (N is the number of class labels and the background) A bounding box regression layer with 4 * N output parameters is the second layer. Layer returns the position of the bounding box of the object in the image. The RPN network trains the recognition network of the objects separately. The weights of a detector network are now used to initialize the RPN (Fast RCNN). This time only the RPN-specific layer weights are fine-tuned. The Fast RCNN [figure 3] detector is tuned using the new matched RPN. Only the specific layers of the detector network are matched, while the common layer weights are determined. [28-31]

3.4. Proposed Work

The proposed algorithm for Tomato leaf detection is briefed below.

Step1. Capture tomato leaf image using digital camera or mobile camera.

Step2. Apply data augmentation methods

Step3. Resize the image

Step4. The resultant image is then subjected to (i) Yolo (version 1 and 3) and (ii) Faster RNN for detection of diseased part of leaf followed by classification using SVM & Random tree.

Step 5. Display the disease detected (outcome of YOLO /Faster RNN), classification accuracy (outcome of SVM and Random Forest), area affected by disease and remedy in the form of use of appropriate pesticides/fertilizer to treat the identified tomato plant diseases.

4. RESULTS

The proposed work has been carried out using the Data set briefed in Table 2.

Table 2. Tomato leaf Dataset used in the proposed work

Tomato leaf Dataset Source	# training images	# testing images
Kaggle(10 directory)	10000	1000
Images from other websites	800	200
Live Images using mobile camera and digital camera	150 Training Images	50

The tomato disease detection outcome of YOLO and Faster RNN are presented in table 3.

Table 3 clearly depicts the improved detection results of YOLO over Faster RNN. YOLO is much

better in detecting the smaller parts and the computation speed is more rather than Faster RCNN. To calculate the accuracy of classification of tomato disease, two methods: RFC (Random Forest Classifier) and SVM (Support Vector Machine) are used. The YOLO v1 one estimated accuracy with Random Forest Classifier (RFC) in the range of 80% -90%. The YOLO v3 showed better results than YOLO v1 in the range of 90%-94% with SVM and more than 97% with RBC. Results of proposed work is better than results as mentioned in [17] with MobileNet, DarkNet-19, Inception v2 and ResNet-101. Results obtained closely match with Darknee results [17]. The proposed work was carried out using dataset not restricted to Kaggle, but also using images from internet and images captured by mobile and digital camera.

Sample snap of mobile app for Late blight disease detected using YOLO v1 are shown in figure 4(a) (uploaded image and binarized image) & (b) for outcome of YOLo v1 and RFC: disease detected, accuracy, area affected in % and remedy in the form of Pesticide name and its usage. Figure 5(a) & (b) showing the outcome using YOLO v3 with RFC for Late blight disease detection.

Table 4 provides details of the 9 tomato diseases and the remedy for the same provided by the agriculture experts. Based on the disease identified the proposed app displays the pesticide name and the quantity of usage as remedy for the farmers. Table 6 and 7 shown snaps of outcome of YOLO v3 and Faster RNN for healthy & 9 types of tomato leaf diseases respectively.

5. DISCUSSIONS

Work was carried out using both live tomato diseased leaf images captured by mobile and digital camera and publicly available tomato leaf images. Data augmentation was applied to increase the dataset size as well as to develop robust model. Tomato disease detection was carried out using YOLO version 1& 3 and Faster RCNN trained on VggNet for tomato disease location and detection followed by identification of disease using SVM and RFC. The combination of YOLO was better compared to Faster RCNN. The RFC results were compared to SVM with YOLO v1 &3 and Faster RCNN. The accuracies were compared with result of [17] and the proposed work proved to be a better solution even with images captured by mobile and digital camera. Since the objective was to develop just-in-time disease detection app for farmers, the app in addition to disease detection shows, the area effected, as well as the remedy in the form of pesticide name and its usage. In terms of computation time, YOLO v1 took the least time for detection in around 45 minutes compared to 90 minutes for YOLO v3 and Fast RNN with 100 minutes.

Table 3. Comparative results of YOLO and Faster RNN with SVM and RFC in proposed work with other models

Proposed Tomato leaf detection methods	Accuracy Range
YOLO v1 with RFC	90%-95%
YOLO v1 with SVM	80%-90%
Faster R-CNN with RFC	75%-89%
Faster R-CNN with SVM	70%-80%
YOLO v3 with RFC	97%-99%
YOLO v3 with SVM	90%-94%
MobileNet [17]	71.1%
DarkNet-19 [17]	74.1%
Inception v2[17]	73%
ResNet-101[17]	76%
Darknet 53[17]	99.1%

There is still lot of scope for further improvements. The proposed work is carried out considering only single image. As part of future work, the work can be extended by considering images of tomato leaves in group as well work on tomato fruits also. The developed app supports only two languages Kannada and English. In this context, app will be explored to support multiple languages.

Figure 4 (a) Snap of uploaded image of Late blight and its binarized image

Figure 4 (b) Snap of Result of YOLO v1 with RFC for Late blight

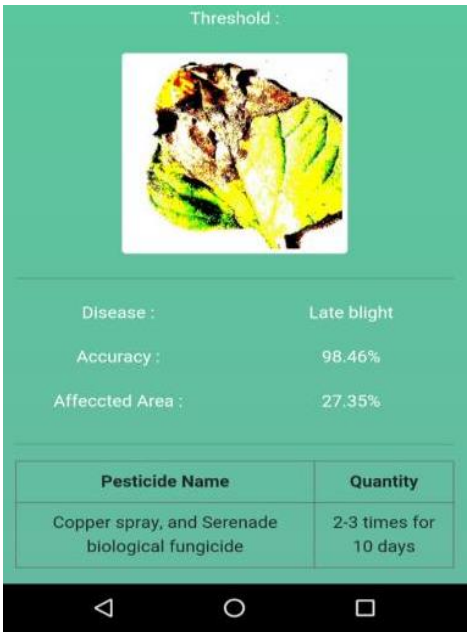


Figure 5 Snap of Result of YOLO v3 with RFC for Late blight image

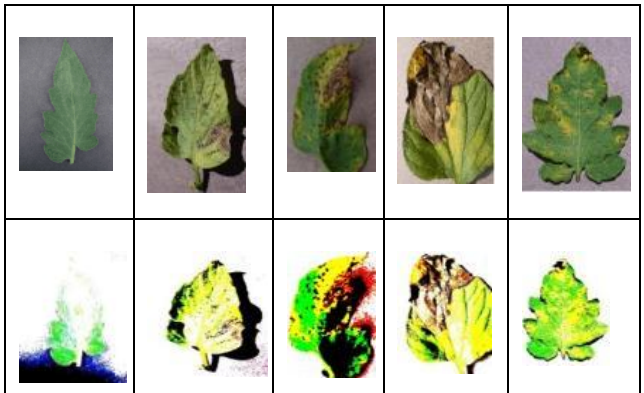
Table 4. Types of Tomato Disease with the proper pesticide and quantity to be used

Type of Tomato Disease	Name of Pesticide	Usage/Quantity
Bacterial Spot	Agrimycin-100	Thrice at 10 days intervals
Early Blight	Combination of copper and mancozeb containing fungicides	2-3 times for 10 days
Late Blight	Copper spray, and Serenade biological fungicide	2-3 times for 10 days
Mosaic Virus	Trisodium Phosphate	90 g/liter of water
Septoria Leaf Spot	Dithane Z-78	0.2%
Target Spot	Monocrotophos	0.05%
Spider Mites	praying plants with a strong jet of water	insecticidal soap
Leaf Mold	Fungicide	0.15%
Yellow Leaf curl Virus	2-3 foliar sprays	Dimethoate (0.05%)

6. CONCLUSION

The main objective of the proposed work was to detect diseased part and to identify types of tomato leaf disease using deep learning techniques. The work was carried out using both publicly available tomato leaf images as well as using images captured by mobile and digital camera for identifying 9 types of tomato leaf diseases. Disease detection of YOLO provided promising results in less computation time compared to Faster RNN. YOLO v3 with RFC resulted in accuracy more that 97%. The work supports both web platforms and the Android application. The developed application is farmer friendly and helps the farmer to identify type of tomato diseases, the affected area, accuracy of detection and the remedy for the disease identified. RFC proved to better than SVM for classification of diseases with all the three models: YOLO v1, v3 and faster RNN. As part of future work, the app functionality will be extended to recognize many more crops diseases as well as support few more languages in addition to existing Kannada and English.

Table 5. Tomato leaf disease detection using YOLO v3 (1st row and 4th row origional images followed by segmented outcome in 2nd and 5th row)



Healthy leaf	Early Blight	Bacterial Spot	Late Blight	Septoria Leaf spot

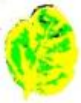
				
				
Leaf Mold	Spider Mites	Target Spot	Yellow Leaf curl	Mosiac Virus

Table 6. Tomato leaf disease detection using Faster RNN (1st row and 4th row origional images followed by segmented outcome in 2nd and 5th row)

				
				
Septoria Leaf spot	Spider Mites	Target Spot	Mosiac Virus	Yellow Leaf curl
				
				
Healthy leaf	Bacterial spot	Early Blight	Late Blight	Leaf Mold

REFERENCES

[1] T. T. Mim, M. H. Sheikh, R. A. Shampa, M. S. Reza and M. S. Islam, "Leaves Diseases Detection of Tomato Using Image Processing," 2019 8th International Conference System Modeling and Advancement in Research Trends (SMART), 2019, pp. 244-249, doi: 10.1109/SMART46866.2019.9117437].

[2] Y. Lu, S. Yi, N. Zeng, Y. Liu and Y. Zhang, "Identification of rice diseases using deep convolutional neural networks," Neurocomputing, 2017, vol 267, 378-384

[3] C. K. Sampoorna, K. Rasadurai , Tomato Leaf Disease Detection using K-Means, SVM Classifier & Neural Networks, International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8 Issue-5, January 2020

[4] J. Ma, K. Du, F. Zheng, L. Zhang, Z. Gong, and Z. Sun, "A recognition method for cucumber diseases using leaf symptom images based on deep convolutional neural network," Computers and Electronics in Agriculture, vol. 154, pp. 18–24, 2018.]

[5] Qimei Wang, Feng Qi , Minghe Sun , Jianhua Qu , and Jie Xue, Identification of Tomato Disease Types and Detection of Infected Areas Based on Deep Convolutional Neural Networks and Object Detection Techniques, Computational Intelligence and NeuroscienceVolume 2019 |Article ID 9142753 | https://doi.org/10.1155/2019/9142753

[6] Geetharamani and J. Arun Pandian, "Identification of plant leaf diseases using a nine-layer deep convolutional neural network," Computers & Electrical Engineering, vol. 76, pp. 323–338, 2019.

[7] Muhammad Hammad Saleem, Johan Potgieter and Khalid Mahmood Arif, "Plant Disease Detection and Classification by Deep Learning" , Plants , 2019, 8(11), 468; https://doi.org/10.3390/plants8110468

[8] DongyiWang , RobertVinson, Maxwell Holmes, Gary Seibe, Avital Bechar, Early Detection of Tomato Spotted Wilt Virus by Hyperspectral Imaging and Outlier Removal Auxiliary Classifer Generative Adversarial Nets (OR-AC-GAN), Shimon Nof3 & YangTao1 Scientific Reports | (2019) 9:4377 | https://doi.org/10.1038/s41598-019-40066-y

[9] Moghadam, P.; Ward, D.; Goan, E.; Jayawardena, S.; Sikka, P.;Hernandez, E. Plant disease detection using hyperspectral imaging. In Proceedings of the 2017 International Conference on Digital Image Computing: Techniques and Applications (DICTA), Sydney, Australia,29 November–1 December 2017; pp. 1–8.

[10] Mohit Agarwal, Abhishek Singh, Siddhartha Arjaria, Amit Sinha, Suneet Gupta,ToLeD: Tomato Leaf Disease Detection using Convolution Neural Network, Procedia Computer Science, Volume 167, 2020, Pages

293-301, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2020.03.225

[11] Nachtigall, L.G.; Araujo, R.M.; Nachtigall, G.R. Classification of apple tree disorders using convolutional neural networks. In Proceedings of the 2016 IEEE 28th International Conference on Tools with Artificial Intelligence (ICTAI), San Jose, CA, USA, 6–8 November 2016; pp. 472–476.

[12] DeChant, C.; Wiesner-Hanks, T.; Chen, S.; Stewart, E.L.; Yosinski, J.; Gore, M.A.; Nelson, R.J.; Lipson,H. Automated identification of northern leaf blight-infected maize plants from field imagery using deep learning. Phytopathology 2017, 107, 1426–1432.

[13] Lu, Y.; Yi, S.; Zeng, N.; Liu, Y.; Zhang, Y. Identification of rice diseases using deep convolutional neural networks. Neurocomputing 2017, 267, 378–384.

[14] Valeria Maeda-Gutiérrez , Carlos E. Galván-Tejada , Laura A. Zanella-Calzada ,José M. Celaya-Padilla , Jorge I. Galván-Tejada , Hamurabi Gamboa-Rosales , Huizilopoztli Luna-García , Rafael Magallanes-Quintanar , Carlos A. Guerrero Méndez and Carlos A. Olvera-Olvera , Comparison of Convolutional Neural Network Architectures for Classification of Tomato Plant Diseases, Appl. Sci. 2020, 10, 1245; doi:10.3390/app10041245

[15] Mohammad Amimul Ihsan Aquil, Wan Hussain Wan Ishak, Evaluation of scratch and pre-trained convolutional neural networks for the classification of Tomato plant diseases Article in IAES International Journal of Artificial Intelligence (IJ-AI) · June 2021 DOI: 10.11591/ijai.v10.i2.pp467-475

[16] Koushik Nagasubramanian, Sarah Jones, Asheesh K. Singh , arti singh, Baskar Ganapathysubramanian, Soumik Sarkar,Explaining hyperspectral imaging based plant disease identification: 3D CNN and saliency maps, 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

[17] Anjanadevi B, Charmila I, Akhil NS, Anusha R. An Improved Deep Learning Model for Plant Disease Detection, International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8 Issue-6, March 2020

[18] Bhuvaneswary, N., S. Prabu, K. Tamilselvan, and K. G. Parthiban. "Efficient Implementation of Multiply Accumulate Operation Unit Using an Interlaced Partition Multiplier." *Journal of Computational and Theoretical Nanoscience* 18, no. 4 (2021): 1321-1326.

[19] Kumar, M. Keerthi, B. D. Parameshachari, S. Prabu, and Silvia liberata Ullo. "Comparative Analysis to Identify Efficient Technique for Interfacing BCI System." In *IOP Conference Series: Materials Science and Engineering*, vol. 925, no. 1, p. 012062. IOP Publishing, 2020.

[20] Naeem, Muhammad Ali, Tu N. Nguyen, Rashid Ali, Korhan Cengiz, Yahui Meng, and Tahir Khurshaid. "Hybrid Cache Management in IoT-based Named Data Networking." *IEEE Internet of Things Journal* (2021).

[21] Le, Ngoc Tuyen, Jing-Wein Wang, Duc Huy Le, Chih-Chiang Wang, and Tu N. Nguyen. "Fingerprint enhancement based on tensor of wavelet subbands for classification." *IEEE Access* 8 (2020): 6602-6615.

[22] Seyhan, Kübra, Tu N. Nguyen, Sedat Akleylek, Korhan Cengiz, and SK Hafızul Islam. "Bi-GISIS KE: Modified key exchange protocol with reusable keys for IoT security." *Journal of Information Security and Applications* 58 (2021): 102788.

[23] Subramani, Prabu, K. Srinivas, R. Sujatha, and B. D. Parameshachari. "Prediction of muscular paralysis disease based on hybrid feature extraction with machine learning technique for COVID-19 and post-COVID-19 patients." *Personal and Ubiquitous Computing* (2021): 1-14.

[24] Rajendrakumar, Shiny, and V. K. Parvati. "Automation of irrigation system through embedded computing technology." In *Proceedings of the 3rd International Conference on Cryptography, Security and Privacy*, pp. 289-293. 2019.

[25] Parameshachari, B. D., H. T. Panduranga, and Silvia liberata Ullo. "Analysis and computation of encryption technique to enhance security of medical images." In *IOP Conference Series: Materials Science and Engineering*, vol. 925, no. 1, p. 012028. IOP Publishing, 2020.

[26] N. Shi, L. Tan, W. Li, X. Qi, K. Yu, "A Blockchain-Empowered AAA Scheme in the Large-Scale HetNet", Digital Communications and Networks, https://doi.org/10.1016/j.dcan.2020.10.002.

[27] Y. Sun, J. Liu, K. Yu, M. Alazab, K. Lin, "PMRSS: Privacy-preserving Medical Record Searching Scheme for Intelligent Diagnosis in IoT Healthcare", IEEE Transactions on Industrial Informatics, doi: 10.1109/TII.2021.3070544.

[28] C. Feng *et al.*, "Efficient and Secure Data Sharing for 5G Flying Drones: A Blockchain-Enabled Approach," IEEE Network, vol. 35, no. 1, pp. 130-137, January/February 2021, doi: 10.1109/MNET.011.2000223.

[29] L. Tan, H. Xiao, K. Yu, M. Aloqaily, Y. Jararweh, "A Blockchain-empowered Crowdsourcing System for 5G-enabled Smart Cities", Computer Standards & Interfaces, https://doi.org/10.1016/j.csi.2021.103517

[30] Z. Guo, Y. Shen, A. K. Bashir, M. Imran, N. Kumar, D. Zhang and K. Yu, "Robust Spammer Detection Using Collaborative Neural Network in Internet of Thing Applications", IEEE Internet of Things Journal, vol. 8, no. 12, pp. 9549-9558, 15 June15, 2021, doi: 10.1109/JIOT.2020.3003802.

[31] Z. Guo, K. Yu, Y. Li, G. Srivastava, and J. C. -W. Lin, "Deep Learning-Embedded Social Internet of Things for Ambiguity-Aware Social Recommendations", IEEE Transactions on Network Science and Engineering, doi: 10.1109/TNSE.2021.3049262.