Hybrid Spiking DenseNet-based Plant Leaf Disease Detection in NB-IoT

T. Satyendra Kumar, S.Vimal

***Abstract* — The Narrow Band-Internet of Things (NB-IoT) is a low-power wide-area Internet of Things method that uses cellular assistance and has gained popularity in smart agricultural applications. The radio-access technique known as NB-IoT is the same as 3GPP in that it supports a wide range of use cases and is associated with the quick deployment of enormous machine-type communication. Future smart farming will benefit greatly from the current advancements in IoT systems, which allow for automated procedures with minimal human interaction. This research uses deep learning to establish a new method for NB-IoT plant leaf disease diagnosis. The NB-IoT system concept is first examined using a framework for plant disease detection. Consequently, a Medav filter is used to do picture preprocessing. The leaf's diseased areas are then divided using a Fully Convolutional Neural Network (FCN). Feature extraction is then carried out. Next, the well-known Hybrid Spiking DenseNet-based method (HySkD-Net), which was created by fusing DenseNet and Deep Spiking Neutral Network (DSNN), is used to detect plant leaf diseases. Additionally, the tested HySkD-Net method for identifying plant leaf diseases shown enhanced accuracy, True Positive Rate (TPR), and True Negative Rate (TNR) of 94.757%, 91.876%, and 92.876%.**

***Index Terms*— Narrow Band IoT, Convolutional Neural Network, Hybrid Spiking DenseNet, Deep Spiking Neutral Network.**

I. Introduction

CURRENTLY, modern sensor technology, artificial intelligence, Internet of Things (IoT), communication networks, as well as big data are combined into intricate cyber-physical networks along with digital clones for various agriculture uses, like living plants, agriculture fields, agriculture products, agriculture buildings, as well as agriculture machines. These applications aim to enable stakeholders and farmers to remotely monitor, control, and coordinate daily farm operations, and improve their decision-making capabilities [2, 32, 33]. NB-IoT is an example of an advanced IoT technology that can handle and modify enormous amounts of data [11]. The NB-IoT segment functions as the tool of wireless communication in the NB-IoT system that is accomplished using mobile operators.

Manuscript received March 10, 2024; revised April 23, 2025.

T. Satyendra Kumar is a Research Scholar in Department of Computational Intelligence of SRM Institute of Science and Technology, Kattankulathur, Tamilnadu, India (email: ts4011@srmist.edu.in)

S.Vimal is an Assistant Professor in Department of Computational Intelligence of SRM Institute of Science and Technology, Kattankulathur, Tamilnadu, India (corresponding author; phone: +91 9841502092; email: vimals@srmist.edu.in).

This module establishes direct connections and is designed to provide sensor nodes with long-range and longer battery life due to its IoT-compatible infrastructure [7]. The 5G-NB-IoT is a standardized cellular-based licensed technology envisioned as a promising technology in comparison to other IoT wide-area technologies such as LoRa as well as SigFox [3]. Communication standards like Long Range Wide Area Networks (LoRaWAN) as well as NB-IoT are normally utilized in smart agriculture in enabling low-power communication and long-range communication amongst gateways and sensors. These standards enhance interoperability [9]. The application of Short-Term Memory (STM) and NB-IoT technology for wireless data transmission successfully reduces time consumption and enhances system reliability [16]. To minimize the input of unnecessary labor in agriculture along with enhance production effectiveness, the NB-IoT technique based on the detection of a wireless network for intelligent agricultural greenhouse plants is normally utilized. Through the emerging Internet of Things technology NB-IoT, the system can conduct remote real-time monitoring and intelligent control of the growing environment of crops in agricultural greenhouse [15].

Sensor nodes gather data from fields or greenhouses in smart agriculture and forward it to a control centre for processing [9]. In a very low wide area network, a wireless communication segment is referred to as a connection interface and uses the NB-IoT technology to fulfil the job of information transmission. Its low broadband, low speed, wide coverage, ultra-low power consumption, and support for large numbers of connections can help address the challenges associated with managing, connecting, and acquiring large amounts of "small data" in environmental monitoring systems [10]. Operators employ smart terminals with specific social and commercial value to remotely monitor the ecological conditions in a greenhouse. A sensor segment realizes a collection of environmental information functions in the greenhouse. The greenhouse environment monitoring platform is a real-time, intelligent, and centralized environment monitoring and management platform. [10]. Agricultural analytics also deploys a disease detection system in the greenhouse by integrating computer vision techniques to analyze the raw images [1]. Precision Agriculture (PA) involves monitoring plants, crops, soil and weather parameters, livestock, etc. [17]. Its objective is to improve crop production in challenging as well as multifaceted surroundings. In recent years, there has been a rise in various plant leaf diseases [35-39], resulting in a decline in the average potential crop yield and food availability [14]. Plant diseases mostly infect the leaf of the plant which affects the growth and quality of crops [1]. Plant diseases pose massive threats to agricultural productivity, necessitating their early detection for effective disease management [12]. To avoid economic losses, early detection and identification of leaf disease is considered as a better solution.

Existing plant disease detection techniques are time-consuming and manual, and frequently essential support to handle the complexity as well as information dynamism [34]. These manual methods do not use heterogeneous data to make better decisions [14]. In real-world situations, due to little contrast among background as well as lesion region, it is problematic to recognize plant diseases. Moreover, a wide range of sizes and types of lesions, and a lot of noise are contained in the image [11]. The integration of digital technologies into agriculture has opened new opportunities and possibilities, revolutionizing the way farmers manage their crops, resources, and operations [18]. To mitigate this problem, Artificial Intelligence of Things (AIoT) systems for the early detection of pest and disease risks in crops are proposed [7]. The basic principle of using the IoT and Deep Learning (DL) multi-models in the context of early plant disease detection is to develop an autonomous system that should be pervasive and unique [14]. In recent years, DL technology has made more progress in the plant disease detection study. DL technology in the face of the user is transparent, the researchers of plant protection and statistics professional level are not high, can automatically extract image features and classification of plant disease spots, and can express original image characteristics, eliminating the traditional image recognition technology of feature extraction and classifier design which requires a lot of work [13]. Furthermore, DL has been utilized to detect diseases in leaves on various plants [11]. Convolutional Neural Network (CNN) is employed to recognize an image, which helps to classify and detect plant diseases or recognize nutrient deficiencies from an image. Recurrent Neural Networks (RNNs) are used for time series data, making them valuable for forecasting and anomaly detection [9].

Identifying and early detection of disease is an important factor for production management to improve economic growth. Deep Dense Convolution Neural network is also one of the techniques that identifies the disease effectively with an accuracy of 89% [36].

This paper establishes a novel algorithm for plant leaf disease detection in NB-IoT by employing DL. Initially, an NB-IoT system model is considered with a plant leaf disease detection framework. Here, image preprocessing is carried out by employing the Medav filter. Subsequently, the disease-affected region is segmented based on FCN. Later, feature extraction is performed as well as features such as Complete Local Binary Pattern (CLBP), Gray-Level Co-Occurrence Matrix (GLCM), and CNN features are extracted. Then, the plant leaf disease detection is accomplished through the proposed HySkD-Net i.e. developed by the fusion of DSNN as well as DenseNet.

The paper's remaining section is as follows. A review of the relevant literature is offered in Section II, followed by a discussion of the suggested system in Section III, experimental results in Section IV, and a conclusion in Section V.

II. Related Work

Khan, F.A et al. [1] proposed the Convolutional neural network approach for environmental monitoring and disease detection of plants in the smart greenhouse based on NB-IoT. This method provided simplified structures, and flexible networking for the performance and was successful in evading overfitting. However, this method was not evaluated in contrast to other existing schemes to determine its superiority.

Ahmed, M.A et al. [2] developed the Long Range (LoRa) Based IoT Platform approach for remote monitoring of large-scale agriculture farms in Chile for NB-IoT. This method can be utilized to gather information locally with no need for accessing the internet as well as encounter large-scale agriculture farms design requirements. Yet, this method was not applied to other applications such as the smart grid as well as the smart cities domain. Popli S et al. [3] established the Adaptive small Cell Position Algorithm (ASPA) technique for green farming using NB-IoT. The approach had a high potential to save infrastructure requirements as well as ensure green farming communication. However, this method suffered from high path loss. Mezei I et al. [4] implemented the Winet system for early disease detection of plants based on NB-IoT. This technique could distinguish between moderate and severe illnesses and consumed little power. However, this approach overlooked the possibility of feeding a bigger dataset into the machine-learning algorithm to precisely determine the probability of disease outbreaks.

Zeng Y.F et al. [5] designed an intelligent irrigation system for rice paddies in Taiwan using NB-IoT. This method can be customized according to user needs and had the ability to correctly control the quantity of water irrigation for saving water without affecting the yield of crops. Yet, the method did not consider big data and neural networks to enhance its sophistication. Chen D et al. [6] devised the Next-generation soil moisture sensor web for high-density soil moisture observation with NB-IoT. This method supported high-density soil moisture observation. Furthermore, this model exhibited high flexibility. However, no low-power IoT communication protocols were considered for a better as well as smarter way to generate high temporal as well as spatial resolution for the geo-spatial sensor web.

Blanco-Carmona P et al. [7] proposed the Artificial Intelligence of Things (AIoT) system to develop a pest and disease risk detection system based on NB-IoT. This method enhanced decision-making and improved efficiency. In addition, the approaches had no false positives. Yet, this method had a high computation complexity. Rashid, R et al. [8] developed the Multi-Model Fusion Network (MMF-Net) for early and smart detection of corn plant leaf diseases using NB-IoT. This framework had great robustness and could attain high classification accuracy even in the case of limited samples. However, internet obtainability as well as online algorithms using edge computing were not included for handling real-time prediction.

The key problems faced by the existing approaches for improving agriculture in NB-IoT are as follows.

- The Winet system [4] can only forecast grapevine downy mildew, and it was futile to include other disease models to enable the control of various diseases, not only in viticulture, to enhance generalizability.
- The key challenge faced by the intelligent irrigation system proposed in [5] was that it did not consider collecting field data with this system and combining it with artificial intelligence, big data, and/or neural networks to improve its sophistication and efficiency.

- The AIoT system [7] was retainable, so it had high flexibility to changes such as new pests, new diseases, the type of crop, the type of terrain, etc. Although, it did not include DL methods to enhance the accuracy of the detection process.
- In [8], MMF-Net was proposed for early detection of corn plant leaf diseases and this technique did not incorporate more advanced algorithms with more diverse datasets to enhance accuracy, efficiency, and processing capabilities.
- Many approaches were proposed for plant leaf detection, nevertheless, low-level visual features are an improved solution for correctness, and they did not offer high-level semantic features for the correct detection of plant leaf disease. Moreover, a captured image might need help using background clutter as well as illumination variation.

III. Propsed Methodology

Farm perception, application, actuators and sensors, and communication network are the four layers that make up IoT-based smart farming mechanisms.

- Farm perception layer: This layer is used for monitoring many parameters like humidity and soil moisture, weather conditions, plant monitoring, water monitoring, and machine status. Several measurement devices, actuators, and sensor nodes are available in farm fields. These nodes fix agro machines, mobile sensor nodes, and sensor nodes. The sensor node is utilized to acquire many measuring parameters, like plant, environment, and soil. Enabling information transmission with very little human intervention is the major goal of this layer.
- Sensors and Actuators Layer: Several measurement devices and sensor nodes are connected in the field like soil sensors, temperature sensors, sensors, light sensors, weather stations, images, humidity, as well as videos for gathering information. The gathered information is transferred to gateways or information-gathering points through wireless or wired communication networks. Monitoring information from the sensor and actuator layer is employed to monitor the growth of the crop, where several parameters like fertilization, pesticides, as well as irrigation.
- Communication Network Layer: To permit the information forward to the application layer from the farm perception layer Such solutions contain wired communication technologies like Ethernet and wireless communication technologies like Bluetooth, ZigBee, 3G/4G, NB-IoT, Wi-Fi, LoRa, and Sigfox, which can employ exchange the information with the field gateways support, which is distributed in a field. Additional types of network equipment are routers, network switches, base station infrastructures, and gateways.
- Application Layer: It handles the complete information that is received from measuring devices and sensor nodes through a communication network layer, as well as analytics, data storage, and visualization for different agriculture parameters like weather conditions, soil quality, and irrigation information. The established information permits an end-user for remotely monitoring as well as controlling farm processes. Therefore, this layer allows farm management and contains decision-making and planning. Fig. 1 exhibits the architecture of the system model.

A. *Proposed Plant Disease Detection Framework*

This research implements a novel algorithm for the detection of plant leaf disease in NB-IoT by employing DL. Initially, a system model of NB-IoT is considered, and plant disease detection is performed. Here, image preprocessing is carried out using a Medav filter [19]. Subsequently, disease-affected regions of the leaf are segmented using an FCN [22]. Later, feature extraction is performed, where features, like CNN [25] features, CLBP [23], and GLCM [24] are extracted. Then, plant leaf disease detection is carried out by the proposed HySkD-Net technique, which is developed by the combination of DSNN [20] and DenseNet [21]. Fig. 1 represents a block diagram of the HySkD-Net technique for plant leaf disease detection in NB-IoT by employing DL.

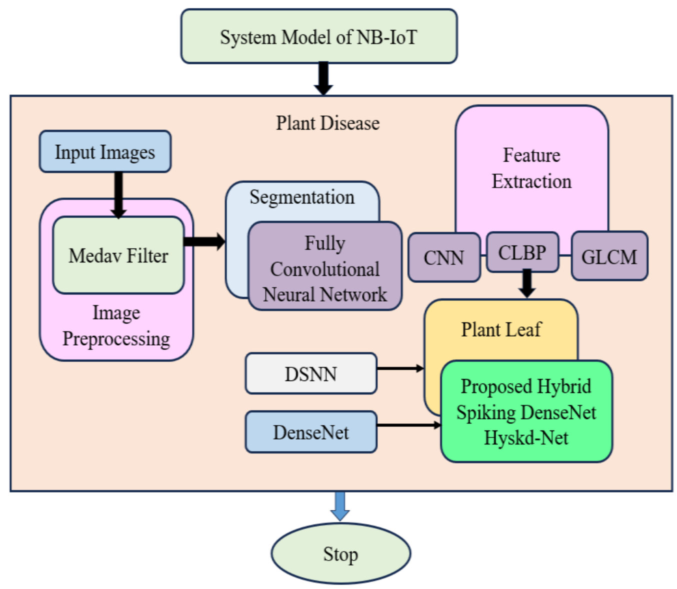


Fig. 1. Proposed HySkD-Net for plant leaf disease detection in NB-IoT.

In this section, the input image is taken from the dataset J, which is expressed as follows,

$$J = \{J_1, J_2, \dots, J_P, \dots, J_W\} \quad (1)$$

where J represents the dataset, J_P denotes P^{th} images in a database, and W signifies the overall image count. Here, the image pre-processing module is applied with the input J_P, and the pre-processing is accomplished by considering the medav filter [19]. The Medav Filter is an integration of adaptive median filters and mean filters, which adjust the mask operation level using noise density. The time complexity of the Medav filter is minimal and hence, it is used here. The Medav filter determines an average value of the mask input first and it is compared to the pixel value. If the average value is less than the considered pixel, then the median value is found and this replaces the pixel value. This process is carried out for each pixel in the image J_P and the output obtained is represented as Architecture of FCN M_f. After image preprocessing, the filtered image M_f is subjected to disease-affected region segmentation, which is carried out with FCN [22]. Segmentation is accomplished to recognize the leaf infected portion.

The process of disease affected region segmentation with FCN algorithm is as follows.

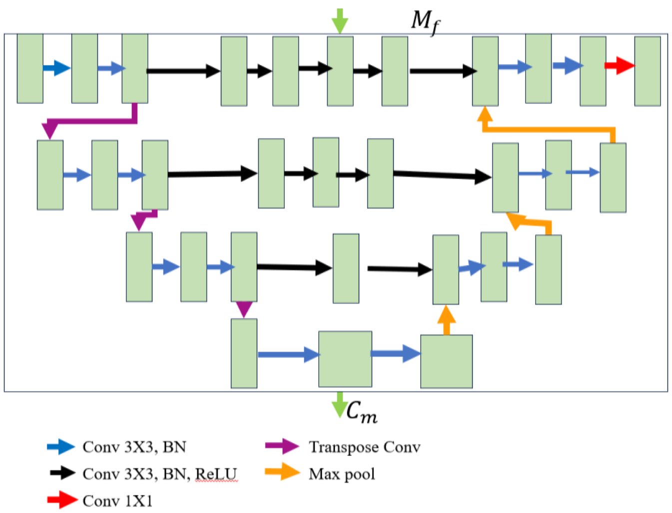


Fig. 2. The Schematic View of FCN Algorithm

The FCN algorithm [22] is developed based on the U-Net algorithm that contains a symmetric encoder or down-sampling and decoder paths or up-sampling, using skip connections amongst them. Here, the U-Net encodes context data, which is applied to the up-sampling path using skip connections that are employed for reestablishing the feature-map restore and for producing a segmentation map. Every down-sampling path step contains dual 3 × 3 convolutions using activation along with batch normalizations and Rectified Linear Unit (ReLU).

A convolution sequence, ReLU, and batch normalization comprise the convolutional blocks or conv blocks. All blocks are motivated by the max-pooling process using the size 2 × 2 as well as a stride is 2. The feature map count is doubled at every down sampling step, as well as this operation is continuous. The bottom layer has a conv block that links an up-sampling path from a down-sampling path. Here, a transposed convolution is employed in the up-sampling path rather than up-convolution. Comparable to a down-sampling path, the 4 × 4 transposed convolution series with the conv block and stride 2 is iterated four times. A last 1 × 1 convolution layer using a softmax activation is utilized to establish a final segmentation map. The skip connections used here are altered by integrating dense blocks to minimize the semantic gap amid down-sampling paths as well as up-sampling paths. Particularly, every dense block layer has 3×3 convolutions using batch normalization as well as ReLU. Every layer integrates its feature maps with those of every previous layer. The filters in every convolutional layer in the dense block are the same as in the adjacent convolutional layer in the down-sampling path. Fig. 2 represents the schematic architecture of the FCN algorithm. The disease-affected region segmentation is designated as C_m.

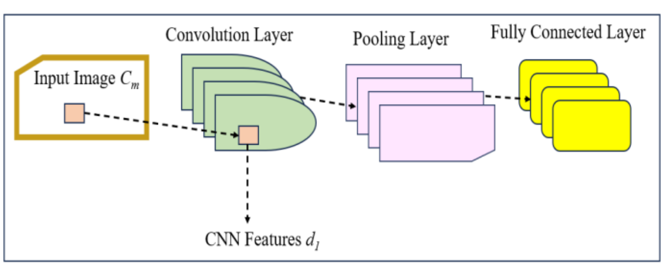


Fig. 3. CNN Feature Architecture

B. *Feature Extraction*

Once the disease-affected region segmentation C_m is completed, feature extraction is carried out, where features like CNN [25] features, CLBP [23], and GLCM [24] are mined and these features are explained as follows,

CNN feature

The CNN feature is generated by applying the disease-affected region-segmented output $C_m$as input to the CNN. The CNN [25] is a multilayered network, which has a fully connected layer, convolution layer, and pooling layer. CNN feature is utilized for image classification as well as object recognition tasks. The conv layer accomplishes a task of mining a feature from the input, as well as the feature dimensionality is reduced by pooling layers. Further, the classification is performed by a fully connected layer. Here, a CNN feature is extracted from a convolution layer. A CNN feature is indicated as d_1 . Fig. 3 portrays CNN feature architecture.

CLBP feature

CLBP [23] is presented to encode local images more correctly in the case of texture classification. Primarily, the difference between neighboring pixels $p_c$of a centre pixel p_l in the segmented image $C_m$is found and is specified as,

$R_c = p_c - p_l$, wherein $R_c$is characterized into many parts as follows,

$$R_c = H_c * I_c \tag{2}$$

were,

$$I_c = |R_c| \tag{3}$$

$$H_c = \begin{cases} 1 & R_c \geq 0 \\ -1 & R_c \leq 0 \end{cases} \tag{4}$$

here, the sign component is specified as $H_c$as well as the magnitude component is represented asI_c. Therefore, CLBP is determined regarding p_l, I_c and H_c is formulated by,

$$CLBP\_I_{c,b} = \sum_{c=0}^{c-1} h(I_c - g)2^c \tag{5}$$

$$h(u, v) = \begin{cases} 1, & u \geq g \\ 0, & u < g \end{cases} \tag{6}$$

here, g represents the mean of $I_c$from an overall image, c specifies the number of neighboring pixels and b is the radius.

$$CLBP\_H_{c,b} = \sum_{c=0}^{c-1} h(H_c - 0)2^c \tag{7}$$

$$CLBP\_e_{c,b} = h(p_l - N_1) \tag{8}$$

here, N_1 represents the mean value of an image. The histogram of $CLBP\_I_{c,b}$, $CLBP\_e_{c,b}$ and $CLBP\_H_{c,b}$ are found and concatenated to yield the CLBP feature designated as d_2.

GLCM features

The GLCM [24] features like contract, dissimilarity, homogeneity, correlation, and energy are determined from the segmented image C_m. These features are discussed beneath,

i) Contract: It measures local differences in an image. Values of very high contrast represent large variances among neighbouring pixel intensities. The subsequent expression is employed to recognize the contrast feature,

$$d_\alpha = -\sum_x \sum_y O_{xy} \log_2 O_{xy} \tag{9}$$

here, O_{xy} specifies the $(x,y)^{th}$ normalized GLCM element, and the contract GLCM feature is designated as d_α.

ii) Dissimilarity: It measures an average variance in intensity among a neighbouring pixel. Values of very high dissimilarity [31] signify better heterogeneity in texture. The dissimilarity feature is indicated as d_β.

$$d_\beta = \sum_{x=0}^{Z-1} \sum_{y=0}^{Z-1} O_{x,y} s(x-y) \tag{10}$$

here, Z denotes the number of gray levels.

iii) Homogeneity: It reflects the distribution closeness elements to the diagonal of GLCM. Values of very high homogeneity represent the elements are integrated along a diagonal, which suggests many uniform textures. Homogeneity is given by,

$$d_\gamma = \sum_x \sum_y \frac{1}{1+(x-y)} O_{xy} \tag{11}$$

here, the homogeneity GLCM feature is represented as d_γ.

iv) Correlation: It measures a linear dependency amongst pixel pairs. The values of very high correlation specify a many predictable textures. The correlation feature is given by,

$$d_\lambda = \sum_x \sum_y O_{xy} \frac{(x-\mu_b)(y-\mu_a)}{\sigma_b \sigma_a} \tag{12}$$

here, μ_b, μ_a denotes the mean, and σ_b, σ_a represents the standard deviation. The correlation GLCM feature is specified as d_λ.

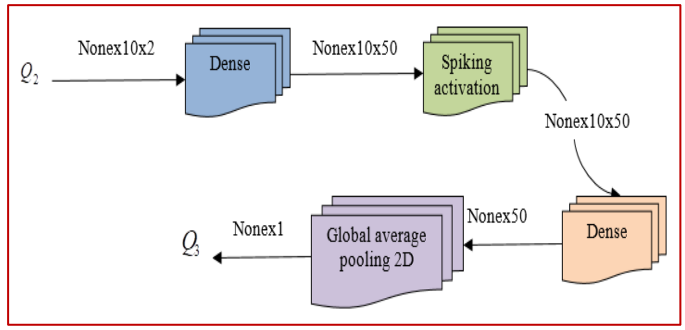


Fig. 5. The structure of DSNN model

v) Energy: It denotes the homogeneity or orderliness of an image. The values of very high energy designate uniform textures. The following expression delineates the energy feature,

$$d_\omega = \sum \sum_x O^2 (x,y) \tag{13}$$

here, the energy GLCM feature is denoted as d_ω. The GLCM features are represented as d_3, which is formed by combining the GLCM features such as contract, dissimilarity, homogeneity, correlation, as well as energy. Then, the feature vector D is determined regarding features like CNN, CLBP, and GLCM that is expressed as follows.

$$D = \{d_1, d_2, d_3\} \tag{14}$$

here, D designates the feature vector.

D. Plant Leaf Disease Detection

The detection of plant disease is effectuated using HySkD-Net using a feature vector D and pre-processed image M_f. The HySkD-Net is formed by the integration of DSNN [20] and DenseNet [21]. At first, a pre-processed image M_f is subjected to DenseNet, and it forms Q_1 as output. Later, the feature vector D as well as the DenseNet output Q_1 are forwarded to the HySkD-Net layer. Here, the integration of the applied inputs is accomplished by utilizing regression modeling using the concept of Fractional calculus [30]. The resultant output Q_2 from a HySkD-Net layer is subjected to the DSNN model, which generates a final plant disease detection output Q_3. Fig. 4 characterizes an architectural diagram of HySkD-Net.

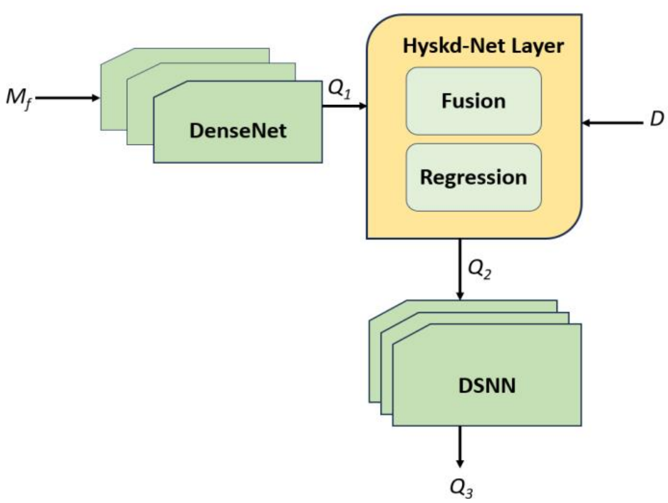


Fig. 4. The Architectural diagram of HySKD Net

DenseNet

This is a widely utilized architecture for the classification among various DL approaches [27]. DenseNet requires very less training parameters when contrasted to another network. It also tackles a vanishing gradient issue using strong feature propagation. Moreover, it needs fewer training parameters and encourages feature recycling. DenseNet contains pooling layers, transition layers, convolutional layers, as well as dense layers. Further, a ReLU activation function is utilized along with a last layer employing the softmax function.

- Convolutional layer: This layer mines the features in an image also forms a filter to input, which results in activation. When a filter is employed in recurrence on an input, the feature map results at different locations. Once a feature map is produced, this is applied with an activation function i.e., ReLU. The filter employed here is normally smaller than an input, a dot product is also employed to operate double entities. Considering the $G \times G$ square neuron component and the dimension of the filter in the convolutional layer be $d \times d$. Let the resulting output dimension be represented as $(g - d + 1) \times (g - d + 1)$. To recognize non-linear input to a unit e_q^r, contributions from a previous layer are combined and is given as follows,

$$e_{bq}^r = \sum_{i=0}^{d-1} \sum_{j=0}^{d-1} \kappa_{ij} \, S_{(b+i)(q+j)}^{r-1} \tag{15}$$

here, $\mathcal{K}$ represents weight, S^r denotes the output produced in the r^{th} layer. The non-linearity function utilized by a convolutional layer is depicted as,

$$Q_1 = \eta(M_f) \tag{16}$$

Here, Q_1 represents the resultant of DenseNet, η signifies the activation function, and M_f specifies the image pre-processing output.

- Maximal pooling layer: This maximum pooling layer reduces a feature map proportionally. Let the feature map dimensions be $k_m \times k_n \times k_e$ designating the width, height, as well as channels. While a filter of size z as well as stride w is applied, the dimension of the maximal pooling layer is as follows,

$$Max_{pooling} = \frac{(k_m - z + 1)}{w} \times \frac{(k_n - z + 1)}{w} \times k_e \tag{17}$$

- Dense layer: This dense layer is deeply combined using the previous layer. Each neuron is associated with further neurons as well as the neurons in this layer obtain the inputs from the neuron of the prior layer and the matrix-vector multiplication is accomplished in this layer and is expressed as,

$$A.\eta = \begin{bmatrix} d_{11} & d_{12} & \ldots\ldots\ldots\ldots & d_{1T} & g_1 \\ d_{11} & d_{12} & \ldots\ldots\ldots\ldots & d_{2p} & g_2 \\ & & . & & \\ & & . & & \\ d_{o1} & d_{o2} & \ldots\ldots\ldots\ldots & d_{oS} & g_S \end{bmatrix} \tag{18}$$

here, A is a matrix with dimensions $o \times S$, g is a matrix with the dimensions $1 \times S$. A variable $\boldsymbol{O}$ is renewed through backpropagation at the training operation time. By means of backpropagation, weights are linked with rS layers found by c^{rS} and the bias by D^{rS} of neural network adjusted over learning rate designated as ρ.

$$c^{rS} = c^{rS} - \rho \times dc^{rS} \tag{19}$$

$$D^{rS} = D^{rS} - \rho \times dD^{rS} \tag{20}$$

Wherein, the chain rule is employed to compute dc and dD that are loss function partial derivatives of C and D, lc and lD are evaluated by,

$$dc^{rS} = \frac{\partial R}{\partial c^{rS}} = \frac{1}{K} dF^{rS} H^{[rS-1]B} \tag{21}$$

$$dD^{rS} = \frac{\partial R}{\partial D^{rS}} = \frac{1}{K} \sum_{a=1}^{K} dF^{rS(a)} \tag{22}$$

$$dH^{[rS-1]} = \frac{\partial R}{\partial H^{[rS-1]}} = P^{rS^B} dF^{rS} \tag{23}$$

$$dF^{rS} = dH^{rS} \times h'(H^{rS}) \tag{24}$$

here, K designates the number of layers, H^{rS}represents linear activation at rS layer, $h'(H^{rS})$specifies the differential of a nonlinear function, as well as F^{rS}specifies nonlinear activation function.

- Transition layer: The transition layer reduces the model complexity as well as reduces the channel sum using the 1×1 convolutional layer. The transition layer also decreases an input height as well as the width through half by employing a filter of stride 2.

Softmax function

A standard non-linear activation is employed for classifying problems in the softmax function, and the activation function is given as,

$$I = z(y \times x + \zeta) \tag{25}$$

The weight is specified as y , bias is ζover input vector $\mathcal{X}$. It is correlated to the output layer of CNN when the possibilities are forecast. This function produces a value for every neuron in the output layer. The Softmax function υ is employed to input σ_b represented by an exponential function j^{σ_b} with d instances and is depicted by,

$$\upsilon(\Lambda)_e = \frac{j^{\sigma_b}}{\sum_{S=1}^{d} j^{\sigma_0}} \tag{26}$$

The loss function employed is called a loss function of binary cross entropy that is expressed by

$$I(p, \zeta) = \frac{1}{k} \sum_{b=1}^{k} . R(i^{(b)}, i^{(b)}) \tag{27}$$

$$R(\hat{\imath}, i) = -(i \times \log\hat{\imath} + (1 - i) \times \log(1 - \hat{\imath})) \tag{28}$$

here, i designates an output class of 1 and $(1 - i)$ is an output class of 0, $\hat{\imath}$ represents a possibility of an output class 1, and $(1 - \hat{\imath})$is the possibility of output class 0. The result output produced from DenseNet is indicated as Q_1.

- HySkD-Net layer: The DenseNet output Q_1 as well as the feature vector D are subjected as inputs to this HySkD-Net layer. In this layer, the fusion of the applied inputs is accomplished by using regression modelling based on the FC [30] concept. FC is a branch of mathematical examination that studies the several ways of defining complex and real number powers of a differentiation and integration operators. The operation of fusion with the FC concept is given below. An output at u^{th} time is measured by employing CNN feature $d_1$and is expressed by,

$$H' = \sum_{i'=1}^{v'} . \sum_{j'=1}^{w'} . d_{1i'j'} * D'_{i'j'} \tag{29}$$

where, $v' \times w'$ signifies feature vector dimension, and D'indicates weight, and H'signifies the u^{th} time output.

An output at $(u-1)^{th}$time is expressed with CLBP feature d_2 and is depicted as,

$$H'_1 = \sum_{i'=1}^{v'} . d_{2_{i'j'}} * D'_{i'j'} \tag{30}$$

where, H'_1 denotes the $(u-1)^{th}$ time output. An output at the $(u-2)^{th}$ time is expressed by employing the GLCM feature d_3, and is expressed as,

$$H'_2 = \sum_{i'=1}^{v'} . \sum_{j'=1}^{w'} . d_{2 \cdot i'j'} * D'_{i'j'} \tag{31}$$

where H'_2 postulates the $(u-2)^{th}$ time output. Then, at $(u-3)^{th}$ instant, the DenseNet output $Q_1$is used to attain the HySkD-Net layer output. Moreover, the fusion is affected using FC [30], and from FC,

$$J'(u'+1)=\alpha' J'(u')+\frac{1}{2}\alpha' J'(u'+1)+\frac{1}{6}(1-\alpha')J'(u'+2)+\frac{1}{24}(1-\alpha')(2-\alpha')J'(u'+3) \quad (32)$$

Substituting outputs at all the instants, the previous equation can be written as,

$$Q_2=\alpha' H'+\frac{1}{2}\alpha' H'_1++\frac{1}{6}(1-\alpha')H'_2+\frac{1}{24}(1-\alpha')(2-\alpha')*Q_1 \quad (33)$$

Now, applying the corresponding values from a prior expression,

$$Q_2=\alpha'\sum_{i'=1}^{v'}.\sum_{j'=1}^{w'}.d_{1\cdot i'j'}*D_{i'j'}+\frac{1}{2}\alpha'\sum_{i'=1}^{v'}.d_{2_{i'j'}}*D_{i'j'}+\frac{1}{6}(1-\alpha')\sum_{i'=1}^{v'}.\sum_{j'=1}^{w'}.d_{3_{i'j'}}*D_{i'j'}+\frac{1}{24}(1-\alpha')(2-\alpha')*\eta(M_f) \quad (34)$$

here, Q_2 characterizes the HySkD-Net layer output, and α signifies constant.

- DSNN Model: A Deep Spiking Neural Network (DSNN) [28] [29] is a type of Artificial Neural Network (ANN) that uses discrete spikes to transmit information. DSNN is a biologically inspired approach for processing information that uses spiking neurons and synapses. DSNNs are alternate to existing deep neural networks for low-power computing. The operation of the DSNN model is illustrated below. Spiking Vector Quantization is used by the neuron in the input layer of this model for generating signed spikes that correspond to negative value or positive value. Let real vector $\vec{h}$represents the input to the neurons array, and L denotes time steps count, Spiking Vector Quantization produces a T signed spike series, $\langle(q_r,n_r):q_r\in[1..\mathrm{len}(\vec{h})],n_r\in\{\pm1\},n\in[1..T]\rangle$, wherein T signifies overall spikes count, which is formed consecutively for T steps, $q_r$signifies the neuron index from which the r^{th} spike fires, $n_r\in\{\pm1\}$refers to the sign of the r^{th}spike. The integrate-and-fire (IF) neuron approach is used by the DSNN. Though the IF neurons not emulate the biological neurons' temporal dynamics, it is ideal to work using rate-coded sensory input wherein spike timings did not play a role. At every j time step, an input spike to the neuron f at the layer vis incorporated below,

$$W_f^v(j)=\delta.\sum_o a_{fo}^{v-1}.\theta_o^{v-1}(j)=\delta.\sum_o a_{fo}^{v-1}.Q_2 \quad (35)$$

where, $\theta_o^{v-1}(j)$ signifies an occurrence of the input spike from an afferent neuron o at time step j. δ specifies the threshold of neuron firing. The term a_{fo}^{v-1} represents synaptic weight, which attaches neuron o from the layer $v-1$. The neuron f incorporates the current input $W_f^v(j)$ into its membrane potential $A_f^v(j)$. Further, the learnable parameter m_f is used to initialize $A_f^v(j)$ as well as the output spike is produced when $A_f^v(j)$is higher than the firing threshold δ. These processes are depicted as below,

$$Q_3=A_f^v(j-1)+W_f^v(j)-\delta.Q_2 \quad (35)$$

$$A_f^v(0)=m_f \quad (36)$$

$$\theta_f^v(j)=\Theta(A_f^v(j)-\delta)\ \text{ with }\ \Theta(k)=\begin{cases}1, & \text{if } k\geq 0\\ 0, & \text{otherwise}\end{cases} \quad (37)$$

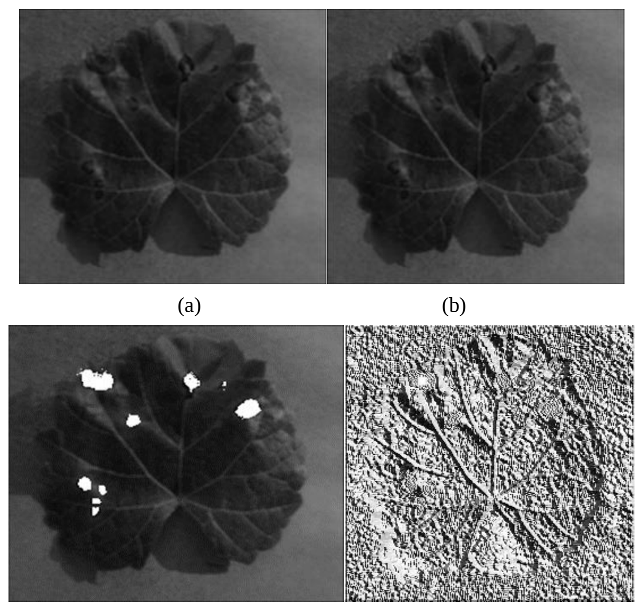
Here, Q_3specifies the resultant of the DSNN model, which is the plant leaf disease detected output. Moreover, the structural diagram of the DSNN model is displayed in Fig. 5.

IV. Results and Discussion

The HySkD-Net mechanism for plant leaf disease detection is realized using a Python tool for execution.

A. *Dataset*

The HySkD-Net approach used the images from the Plant Village-Dataset [26]. The Plant Village dataset is a collection of 54,303 leaf images, both unhealthy and healthy, and it contains 38 types of disease and species. The dataset was introduced by David P. Hughes and Marcel Salathe to help develop mobile disease diagnostics. Image results are attained during the estimation of the HySkD-Net algorithm is portrayed in Fig. 6.



(a) (b) (c) (d)

Fig. 6 Experimental outcomes of the HySkD-Net (a) input image (b) pre-processed image (c) segmented image (d) CLBP feature extracted image

B. *Evaluation Metrics*

The HySkD-Net technique is assessed by considering evaluation measures like TPR, TNR, as well as accuracy

Accuracy is defined as the count of correctly performed plant leaf disease detections to the overall plant leaf images and is represented as,

$$Accuracy=\frac{O+P}{O+P+U+W} \quad (38)$$

here, U postulates false positive, P represents true negative, W specifies false negative, as well as O stipulates true positive.

TPR is the ratio of diseased leaf images, which are accurately detected as positive to a total diseased leaf images is called TPR and it is given as,

$$TPR = \frac{O}{O+W} \tag{39}$$

TNR is diseased leaf images correct negative test results to the overall normal leaf images is termed TNR and is given by,

$$TNR = \frac{P}{P+U} \tag{40}$$

F1Score is the harmonic mean of TPR (Recall) and TNR(Precision) used to assess the performance of the classification machine learning model.

$$F_1Score = \frac{2 \times TPR \times TNR}{TPR+TNR} \tag{41}$$

C. Comparative Assessment

In this case, comparative approach used for the assessment of HySkD-Net are the winet system [4], intelligent irrigation system [5], AIoT [7], and MMF-Net [8]. The efficacy of HySkD-Net is examined using the available techniques for plant leaf disease detections with varying learning set and k-value.

With Learning Set

Performance evaluation and estimation of the various methods of plant disease detection is done and recorded in Table I – Table III. The respective graphical representation is depicted in fig. 7-9. Proposed HySkD-Net method shows better performance compared to other detection methods. When compared with mostly used method Intelligent Irrigation System (IIS) alone, 7.591% enhancement is recorded for 90% Learning Set.

TABLE I
ACCURACY ESTIMATION BASED ON LEARNING SET

% Learning Set	Plant Disease Detection Methods and its accuracy in %				
	Winet system	IIS	AIoT	MMF-Net	Proposed HySkD-Net
60%	77.8767	79.776	82.8768	84.877	87.765
70%	78.765	80.876	83.9888	85.8787	86.8766
80%	80.7865	81.8767	84.8776	86.872	90.766
90%	81.7565	83.877	85.7866	88.876	90.768

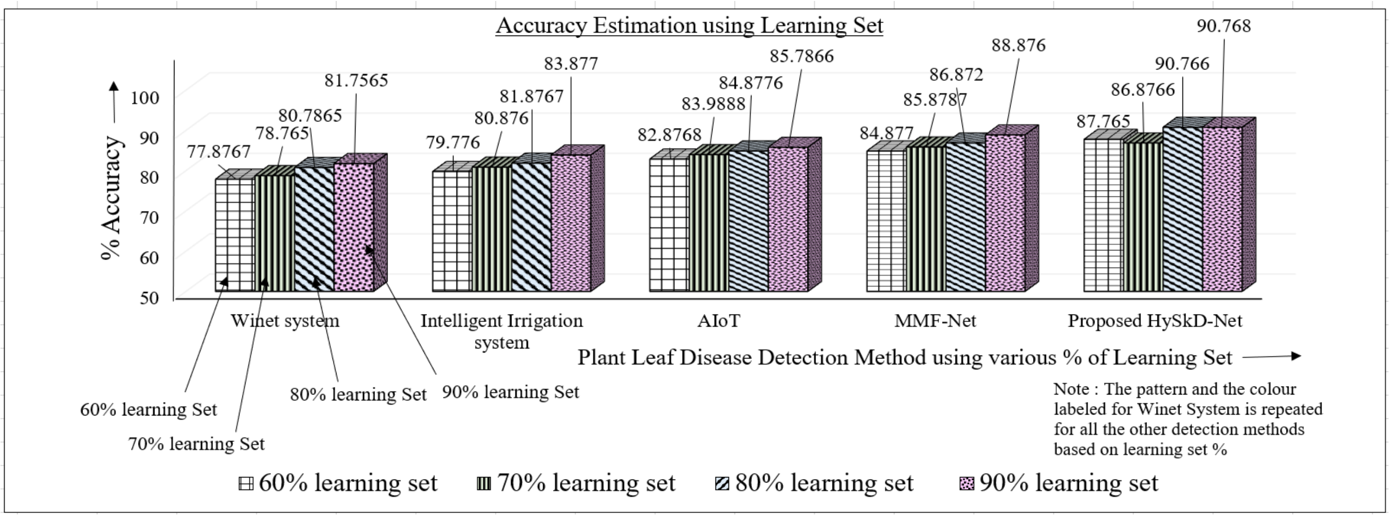


Fig. 7. Accuracy Estimation for Learning Set

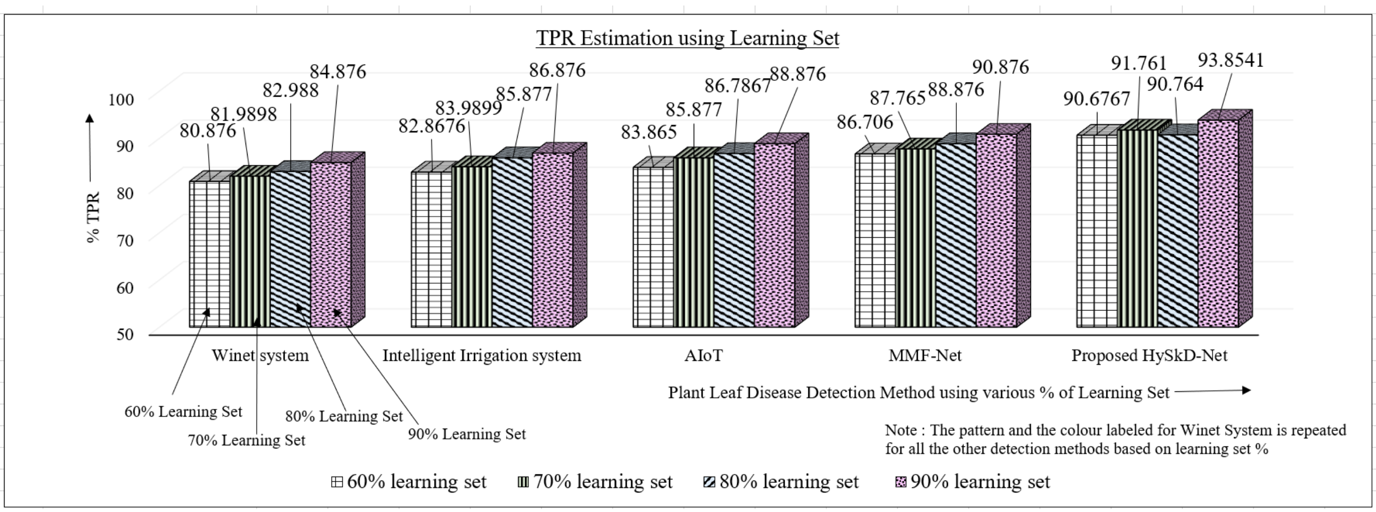


Fig. 8. TPR Estimation using Learning Set

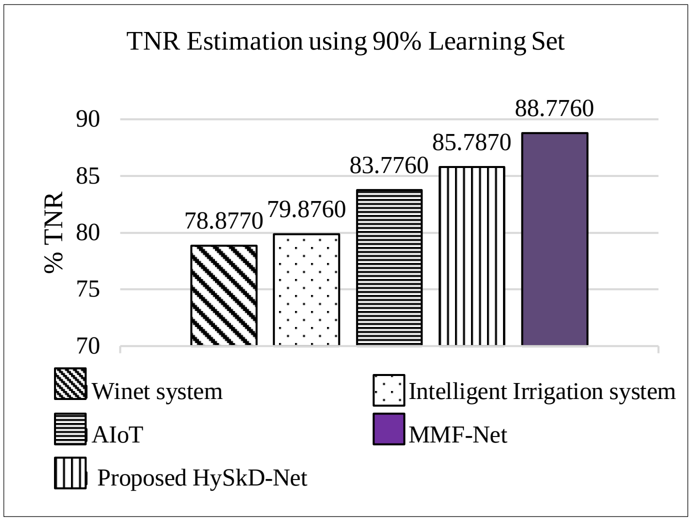


Fig. 9. TNR Estimation using Learning Set

With k-value

Further k-value is used to analyze the performance of plant disease detection methods and the estimated performance obtained is recorded in Table IV – Table VI. The respective graphical representation is depicted in fig. 10-12. Proposed HySkD-Net method shows better performance compared all the detection methods. Accuracy is improved by 2.36% and TPR by 3.43% in proposed HySkD-Net when compared with Intelligent Irrigation System and Winet System respectively.

TABLE IV
ACCURACY EFFICACY BASED ON K-VALUE

K-Value	Plant Disease Detection Methods and its accuracy in %				
	Winet system	IIS	AIoT	MMF-Net	Proposed HySkD-Net
5	79.7866	82.887	84.766	86.6765	88.7656
6	80.766	83.877	85.6755	87.7565	89.8767
7	81.8776	85.7565	87.7656	89.765	92.765
8	83.8776	86.7676	88.7565	90.7676	92.876

TABLE V
TPR EFFICACY BASED ON K-VALUE

K-Value	Plant Disease Detection Methods				
	Winet system	IIS	AIoT	MMF-Net	Proposed HySkD-Net
5	82.7866	84.776	85.766	87.7566	90.8776
6	83.9878	85.876	86.7676	88.8767	91.8678
7	85.7656	86.8767	88.876	90.7676	93.8676
8	86.6765	88.765	89.876	91.7868	94.7565

TABLE VI
TNR EFFICACY BASED ON K-VALUE

Plant Disease Detection Methods	TNR estimation for K-Value 8
Winet system	80.7550
IIS	83.7760
AIoT	86.7868
MMF-Net	87.7650
Proposed HySkD-Net	91.8760

D. Performance analysis of proposed method

The efficacy of HySkD-Net is examined with respect to the available techniques for the plant leaf disease detection with varying learning-set and the k-value. With Learning Set, an estimation of HySkD-Net is accomplished using learning set that is displayed in Table VII and fig. 13. Accuracy is estimated using proposed HySkD-Net methodology. For 90% learning set using epochs, like 10, 20, 30, and 40 is used. An epoch consists of passing a dataset through the algorithm completely. Each Epoch consists of many weight update steps. To optimize the learning process, gradient descent is used, which is an iterative process. It improves the internal model parameters over many steps and not at once Accuracy, TPR and TNR has improved. Also, the evaluation of HySkD-Net using k-value is portrayed in Table VIII and fig. 14.

The evaluation based on accuracy, TPR and TNR of HySkD-Net shows better performance for all Epochs used for Learning set and k-value.

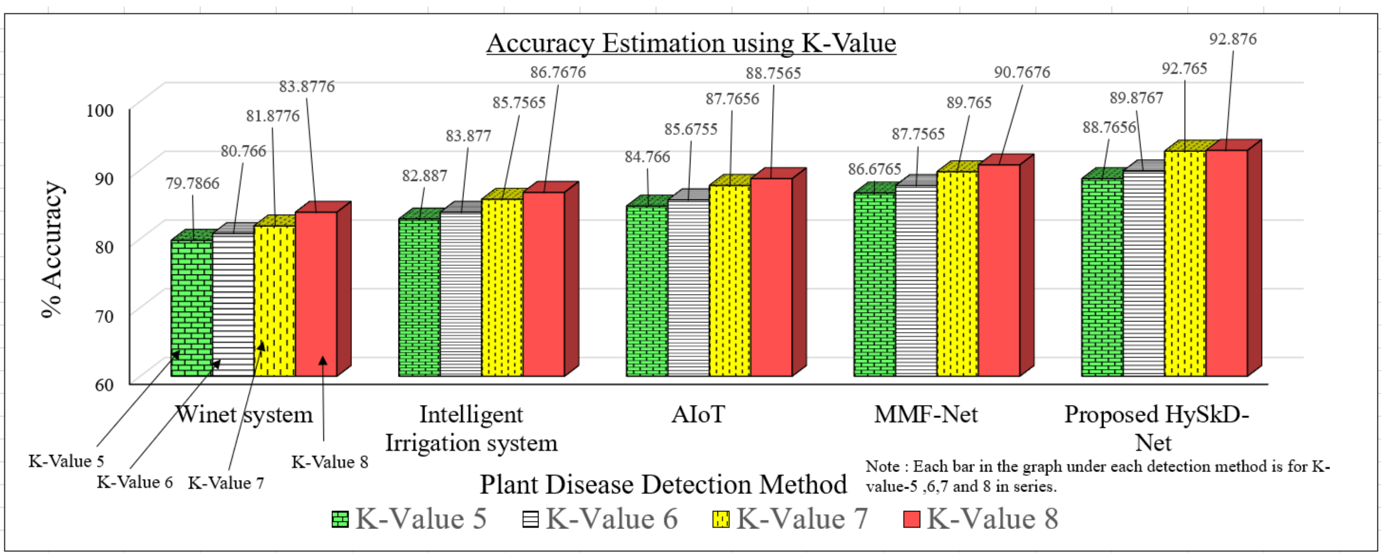


Fig. 10. Accuracy Efficacy for K-Value

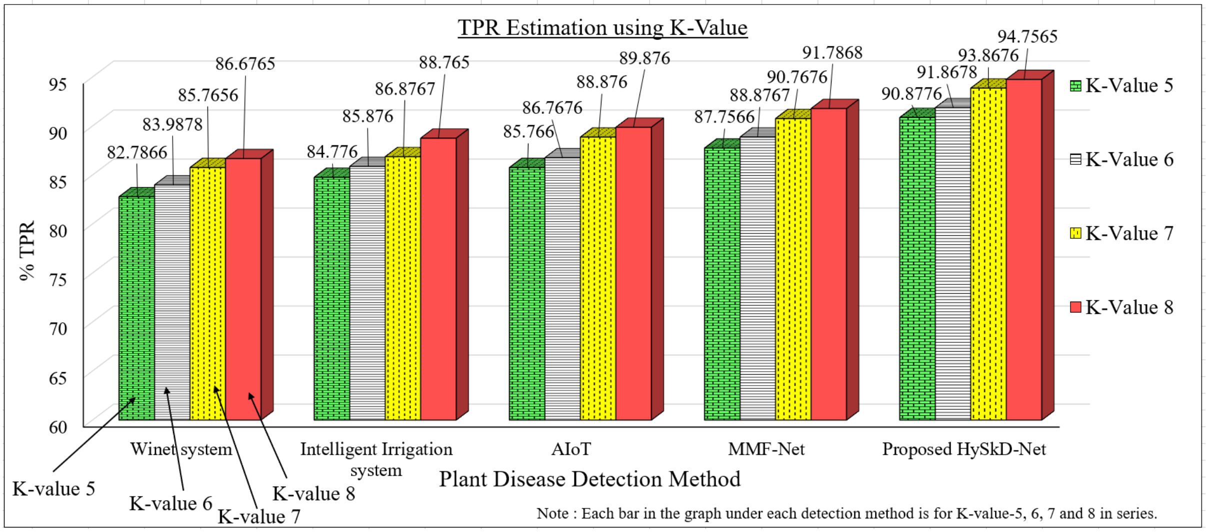


Fig. 11. TPR Estimation using K-Value

E. Comparative analysis of the proposed method

The efficacy of HySkD-Net is assessed based on metrics like TNR, accuracy, and TPR and F1score. Table IX and fig. 15 shows the TPR, accuracy, TNR of the HySkD-Net compared with other techniques using a learning set of 90% and a k-value 8. The accuracy obtained by the approaches like Winet system, Intelligent Irrigation System, AIoT, MMF-Net, and HySkD-Net is 83.878%, 86.768%, 88.757%, 90.768%, and 92.876%. Likewise, a TPR of 86.677%, 88.766%, 89.876%, 91.787%, and 94.757% is computed by the approaches such as winet system, intelligent irrigation system, AIoT, MMF-Net, and HySkD-Net. Moreover, the TNR value recorded by the winet system is 80.755%, the intelligent irrigation system is 83.776%, AIoT is 86.787%, the MMF-Net is 87.765%, also, the HySkD-Net is 91.876%. F1 square is calculated using equation (41). The developed HySkD-Net attained improved classification because of the FCN algorithm, and the fusion of DenseNet along with DSNN in HySkD-Net enabled better plant leaf disease detection, by reducing the error.

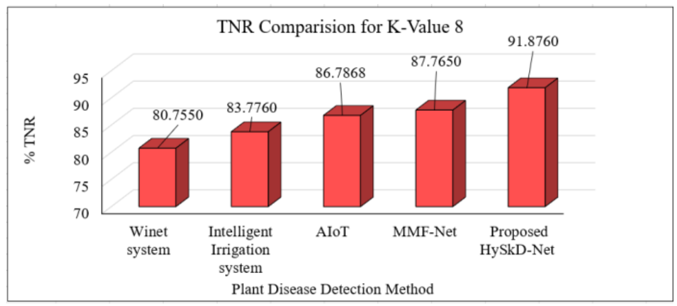


Fig. 12. TNR Estimation using K-Value

TABLE VII
COMPARATIVE ANALYSIS OF THE PROPOSED METHOD FOR 90% LEARNING SET

% Learning set	10 - Epochs			10 - Epochs			10 - Epochs			10 - Epochs		
	Accuracy	TPR	TNR	Accuracy	TPR	TNR	Accuracy	TPR	TNR	Accuracy	TPR	TNR
60%	82.8987	85.7656	80.8767	84.8676	87.7876	82.7868	85.7650	88.8766	83.6765	87.7650	90.6757	85.7650
70%	83.8770	86.6550	81.7868	85.7660	886757.0000	83.9800	86.8767	89.654	84.76	88.8766	91.7865	87.7565
80%	84.7877	87.6755	83.8767	86.7660	89.7650	84.8760	87.7866	90.6757	85.7660	90.7868	92.7650	88.7656
90%	85.7660	88.7868	84.8760	87.8676	90.6650	85.6766	88.7660	91.7866	86.8767	91.7650	93.6544	89.7650

TABLE VIII
COMPARATIVE ANALYSIS OF THE PROPOSED METHOD FOR 90% LEARNING SET

K-Set Values	10 - Epochs			10 - Epochs			10 - Epochs			10 - Epochs		
	Accuracy	TPR	TNR	Accuracy	TPR	TNR	Accuracy	TPR	TNR	Accuracy	TPR	TNR
5	83.8770	86.7656	82.6757	84.7876	87.7687	84.8767	86.6757	88.7660	85.7876	88.7556	90.8376	87.7660
6	84.9878	87.8760	83.7650	85.7760	88.7876	85.7868	87.8676	89.876	86.7867	89.8767	91.8676	88.7656
7	85.7876	88.7565	85.7676	86.7650	89.8676	86.7565	88.7868	91.7676	87.8760	92.7650	83.8616	90.8767
8	86.5540	89.6554	86.7865	87.8767	91.7660	87.8760	90.8767	92.7866	88.7660	92.8760	94.7565	91.8760

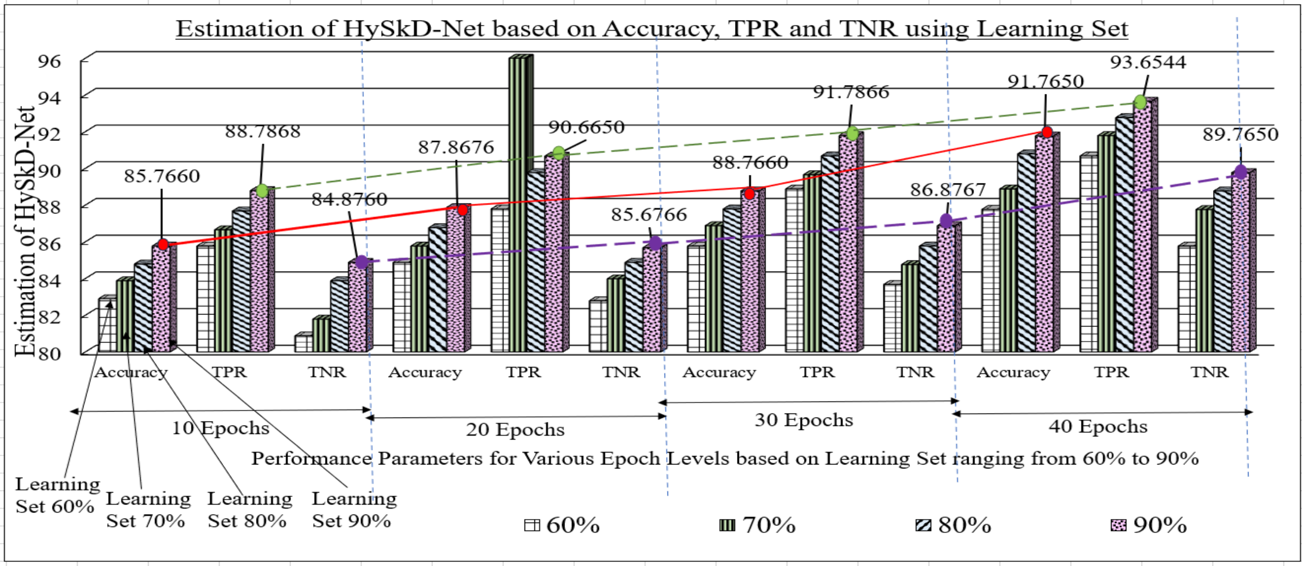


Fig. 13. Comparison Chart of proposed method performance estimation based on Learning Set Variation

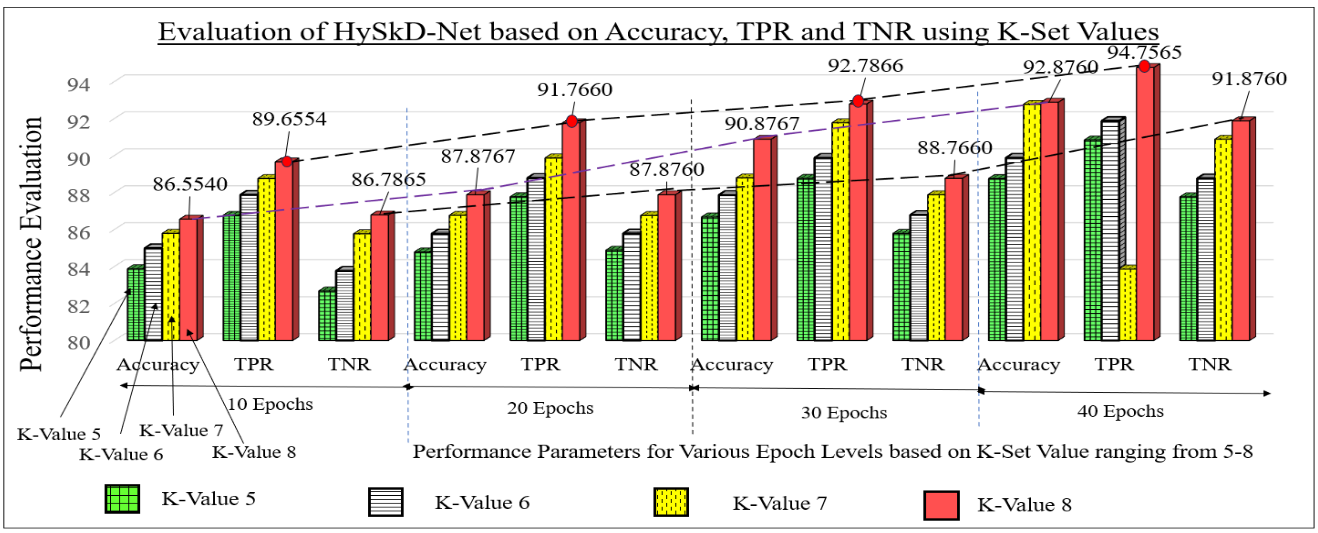


Fig. 14. Comparison Chart of proposed method performance estimation based on K-Set Variation

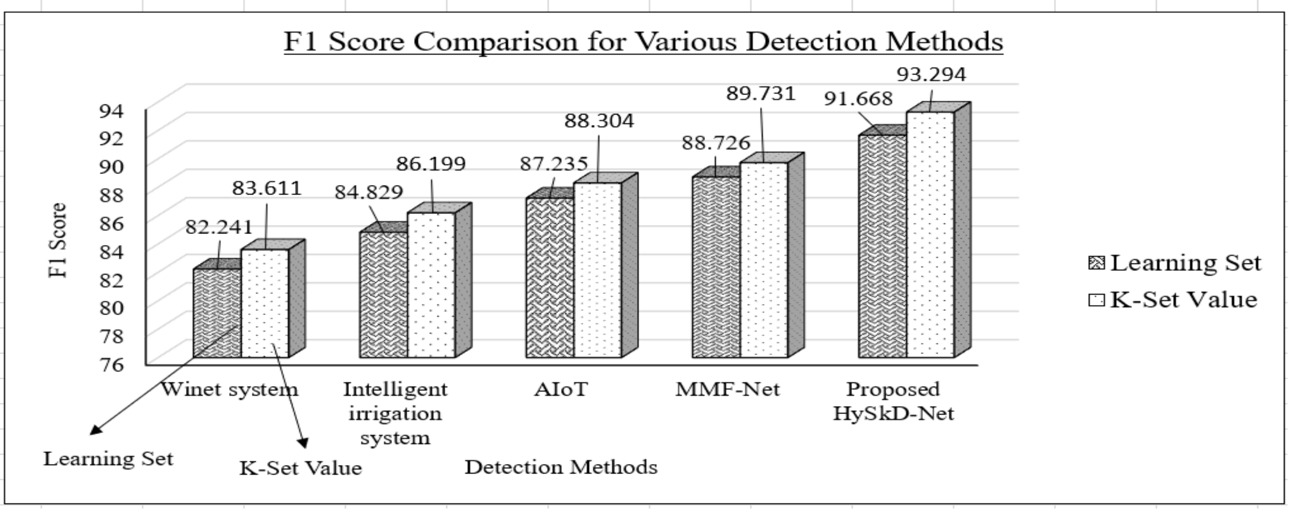


Fig. 15. Comparison Chart of F1 Score of Learning Set Variation and K-set Variations for various methods

TABLE IX
COMPARATIVE ANALYSIS OF F1 SCORE USING THE PROPOSED METHOD

Variations	Metrics	Methods				
		Winet system	Intelligent Irrigation system	AIoT	MMF-Net	Proposed HySkD-Net
	F1Score	82.24117	84.8294	87.2353	88.7263	91.6683
	F1Score	83.6113	86.1988	88.3045	89.7309	93.2943

V. CONCLUSION

To promote healthy crop production, the primary and accurate classification and identification of illnesses are essential. This article suggests a unique method for NB-IoT plant leaf disease detection using DL. First, the plant disease detection method is developed, and the NB-IoT system model is taken into consideration. Image preprocessing is primarily done with a Medav filter. Afterwards, FCN is used to segment the leaf's diseased areas. The next step is featuring extraction, which extracts features like CNN, CLBP, and GLCM features. The suggested HySkD-Net method, which is created by combining DSNN with DenseNet, is then used to detect plant leaf diseases. Furthermore, the effectiveness of the recently presented HySkD-Net technique is assessed using the Plant Village database, considering various k-value values in addition to the training set. Additionally, the enhanced accuracy, TPR, and TNR of the developed HySkD-Net method for plant leaf disease detection were 92.876%, 94.757%, and 91.876%, respectively.

In the future, the image augmentation technique will be utilized for more improvement in the recently introduced HySkD-Net approach. The other feature extraction techniques will also be employed by the recently developed HySkD-Net method. In the future, we plan to expand our study by utilizing other CNNs that have already received multi-classification training and expand the scope of our proposed model to include more plant diseases. The developed tool will be updated to improve the accuracy of the treatment section and incorporate weather and crop circumstances to better diagnose plant diseases.

REFERENCES

[1] Khan, F.A., Ibrahim, A.A. and Zeki, A.M., “Environmental monitoring and disease detection of plants in the smart greenhouse using internet of things”, Journal of Physics Communications, vol. 4, no.5, 2021

[2] Ahmed, M.A., Gallardo, J.L., Zuniga M.D., Pedraza M.A., Carvajal, G., Jara N. and Carvajal R., “LoRa based IoT platform for remote monitoring of large-scale agriculture farms in Chile. Sensors, vol.22, no.8, pp.2824, 2022

[3] Popli, S., Jha, R.K. and Jain, S., “Adaptive Small Cell position algorithm (ASPA) for green farming using NB-IoT”, Journal of Network and Computer Applications, vol. 173, pp.102841, 2021

[4] Mezei, I., Lukić, M., Berbakov, L., Pavković, B. and Radovanović, B., “Grapevine downy mildew warning system based on NB-IoT and energy harvesting technology”, Electronics, vol.11, no.3, pp.356, 2022

[5] Zeng, Y.F., Chen, C.T. and Lin, G.F., “Practical application of an intelligent irrigation system to rice paddies in Taiwan”, Agricultural Water Management, vol. 280, pp.108216, 2023

[6] Chen, D., Chen, N., Zhang, X., Ma, H. and Chen, Z., “Next-generation soil moisture sensor web: High-density in situ observation over NB-IoT”, IEEE Internet of Things Journal, vol.8, no.17, pp.13367-13383,2021

[7] Blanco-Carmona, P., Baeza-Moreno, L., Hidalgo-Fort, E., Martín-Clemente, R., González-Carvajal, R. and Muñoz-Chavero, F., “AIoT in Agriculture: Safeguarding Crops from Pest and Disease Threats. Sensors”, vol. 23, no.24, pp.9733,2023

[8] Rashid, R., Aslam, W. and Aziz, R., “An Early and Smart Detection of Corn Plant Leaf Diseases Using IoT and Deep Learning Multi-Models”, IEEE Access, 2024

[9] Ahmed, A., Parveen, I., Abdullah, S., Ahmad, I., Alturki, N., and Jamel, L., “Optimized Data Fusion with Scheduled Rest Periods for Enhanced Smart Agriculture via Blockchain Integration”, IEEE Access, 2024

[10] Chen, X., Wang, X. and Shen, H., “Design of greenhouse environment monitoring system based on NB-IoT and edge computing”, In Proceedings of 2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), IEEE, pp.1319-1324, March 2021

[11] Moses, M.L., Tharun, K.T., Karthikeyan, S., Prasath, R., Ramya, P. and Mounaguru, P., “IoT-based Crop Health Monitoring and Controlling Smart Agriculture System in Machine Learning”, International Research Journal of Engineering and Technology, vol. 5, Issue, 3, pp. 333-339, 2023

[12] Maurya, R., Mahapatra, S. and Rajput, L., “A Lightweight Meta-Ensemble Approach for Plant Disease Detection Suitable for IoT-Based Environments”, IEEE Access, vol.12, pp.28096-28108, 2024

[13] Li, L., Zhang, S. and Wang, B., “Plant disease detection and classification by deep learning—a review”, IEEE Access, vol. 9, pp.56683-56698,2021

[14] Rashid, R., Aslam, W. and Aziz, R., “An Early and Smart Detection of Corn Plant Leaf Diseases Using IoT and Deep Learning Multi-Models” IEEE Access, 2024

[15] Ye, H., Yang, Y. and Zhu, L., “A wireless network detection and control system for intelligent agricultural greenhouses based on NB-IoT technology”, In Proceedings of Journal of Physics: Conference Series, IOP Publishing, vol. 1738, no. 1, pp. 012058, 2021

[16] Gopan, Y., “Smart Agriculture: IoT-Enhanced Plant Irrigation through Soil Moisture Analysis”, International Journal of Innovative Science and Research Technology, vol. 9, Issue 1, 2024

[17] Singh, D.K., Sobti, R., Jain, A., Malik, P.K. and Le, D.N.,“LoRa based intelligent soil and weather condition monitoring with the internet of things for precision agriculture in smart cities”, IET Communications, vol. 16, no.5, pp.604-618,2022

[18] Karunathilake, E.M.B.M., Le, A.T., Heo, S., Chung, Y.S. and Mansoor, S., “The path to smart farming: Innovations and opportunities in precision agriculture”, Agriculture, vol.13, no.8, pp.1593, 2023

[19] Gupta, S. and Roy, S., “Medav filter—filter for removal of image noise with the combination of median and average filters”, In Recent Trends in Signal and Image Processing: ISSIP 2017, Springer Singapore, pp. 11-19, 2021

[20] Zhu, Z., Lu, S., Wang, S.H., Gorriz, J.M. and Zhang, Y.D., “DSNN: a DenseNet-based SNN for explainable brain disease classification”, Frontiers in Systems Neuroscience, vol.16, pp.838822, 2022

[21] Hemalatha, J., Roseline, S.A., Geetha, S., Kadry, S. and Damaševičius, R., “An efficient dense net-based deep learning model for malware detection”, Entropy, vol.23, no.3, pp .344, 2021

[22] Penso, M., Moccia, S., Scafuri, S., Muscogiuri, G., Pontone, G., Pepi, M. and Caiani, E.G., “Automated left and right ventricular chamber segmentation in cardiac magnetic resonance images using dense fully convolutional neural network”, Computer Methods and Programs in Biomedicine, vol. 204, pp.106059, 2021

[23] Guermoui, M. and Mekhalfi, M.L., “A sparse representation of complete local binary pattern histogram for human face recognition”, arXiv preprint arXiv:1605.09584,2016

[24] Senthilkumar C. and Gnanamurthy, R.K., “A Fuzzy clustering-based MRI brain image segmentation using back propagation neural networks”. Cluster Computing, vol. 22, no.5, pp.12305-12312,2019

[25] Su, R., Liu, T., Sun, C., Jin, Q., Jennane, R. and Weid, L., “Fusing convolutional neural network features with hand-crafted features for osteoporosis diagnoses”, Neurocompting, vol. 383, no. 14, pp. 300-309, 2019

[26] Plant Village-Dataset is available at https://github.com/spMohanty/PlantVillage- dataset/tree/master/raw/color”, accessed on March 2024

[27] Vulli, A., Srinivasu, P.N., Sashank, M.S.K., Shafi, J., Choi, J. and Ijaz, M.F., “Fine-tuned DenseNet-169 for breast cancer metastasis prediction using FastAI and 1-cycle policy”, Sensors, vol.22, no.8, pp.2988, 2022

[28] O'Connor, P. and Welling, M., “Deep spiking networks”, arXiv preprint arXiv:1602.08323, 2016

[29] Wu, J., Chua, Y., Zhang, M., Yang, Q., Li, G. and Li, H., “Deep spiking neural network with spike count based learning rule”, In Proceedings of 2019 International Joint Conference on Neural Networks, IEEE, pp. 1-6, July 2019

[30] Bhaladhare, P.R. and Jinwala, D.C., “A clustering approach for the-diversity model in privacy preserving data mining using fractional calculus-bacterial foraging optimization algorithm”, Advances in Computer Engineering, vol. 2014, article ID 396529, 2014

[31] Iqbal, N., Mumtaz, R., Shafi, U. and Zaidi, S.M.H., “Gray level co-occurrence matrix, texture-based crop classification using low altitude remote sensing platforms”, Peer Journal of Computer Science, vol.7, 2021

[32] Adedamola O. Adedoja, Pius A. Owolawi, Temitope Mapayi, and Chunling Tu, "Intelligent Mobile Plant Disease Diagnostic System Using NASNet-Mobile Deep Learning," IAENG International Journal of Computer Science, vol. 49, no.1, pp216-231, 2022

[33] Manuel Chad Agurob, Amiel Jhon Bano, Immanuel Paradela, Steve Clar, Earl Ryan Aleluya, and Carl John Salaan, "Autonomous Vision-based Unmanned Aerial Spray System with Variable Flow for Agricultural Application," IAENG International Journal of Computer Science, vol. 50, no.3, pp1058-1073, 2023

[34] Panduranga Vital Terlapu, U D Prasan, T. Ravi Kumar, Vijaya Bendalam, Sasibhushana Rao Pappu, M. Jayanthi Rao, Maddula Ratna Mohitha, and Jayaram. D, "Rice Category Identification through Deep Transfer Learning Features and Machine Learning Classifiers: An Intelligent Approach," IAENG International Journal of Computer Science, vol. 51, no. 7, pp765-784, 2024

[35] R. Dwivedi, T. Dutta and Y. -C. Hu, "A Leaf Disease Detection Mechanism Based on L1-Norm Minimization Extreme Learning Machine," in IEEE Geoscience and Remote Sensing Letters, vol. 19, pp. 1-5, 2022, Art no. 8019905, doi: 10.1109/LGRS.2021.3110287

[36] Amit Kumar Pandey, Dhyanendra Jain, Tarun Kumar Gautam, Jitendra Singh Kushwah, Saurabh Shrivastava, Rajeev Sharma, and Prashant Vats, "Tomato Leaf Disease Detection using Generative Adversarial Network-based ResNet50V2," Engineering Letters, vol. 32, no. 5, pp965-973, 2024

[37] Jana S, Thilagavathy S. D., Shenbagavalli S. T., Srividhya G, Prasad V. S. G and Hemavathy R. (2023). Plant Leaf Disease Prediction Using Deep Dense Net Slice Fragmentation and Segmentation Feature Selection Using Convolution Neural Network. International Journal of Intelligent Systems and Applications in Engineering, 11(6s), 76–85. https://ijisae.org/index.php/IJISAE/article/view/2830

[38] Raja and M. Karthikeyan, "Automated Plant Leaf Disease Classification using Artificial Algae Algorithm with Deep Learning Model," *2023* International Conference on Sustainable Computing and Data Communication Systems (ICSCDS), Erode, India, 2023, pp. 162-167, doi: 10.1109/ICSCDS56580.2023.10104602

[39] R. Pillai, N. Sharma, R. Gupta and A. Sharma, "Classification of Plant Diseases using DenseNet 121 Transfer Learning Model," *2023* International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI), Chennai, India, 2023, pp. 1-6, doi: 10.1109/ACCAI58221.2023.10200401

Authors

Mr.T. Satyendra Kumar is a research scholar currently pursuing part-time research scholar at SRM Institute of Science and Technology he had completed his MCA in Sree Vidyanikethan Engineering College, Rangampeta, Tirupati, JNTUH. Based upon his graduation he has completed his Masters of Engineering from Madanaplli Institute of Technology & sciences (MITS), Madanaplli, JNTUA University. Further honing his skills and deepening his understanding of advanced concepts within his field of interest. With a keen focus on Internet of Things, his research efforts are directed towards exploring innovative methodologies, algorithms, and applications in this domain. His academic pursuits reflect a strong commitment to advancing the field, leveraging technology, and contributing to the wider scientific community. He had Published 08 Papers in International Journal and 2 Conference Papers. He had attended 17 webinars and FDP’s. He had filled 2 Innovation Patents. He is having 15 years of experience in teaching. His areas of Interest are Internet of Things, Big data in software engineering, Software Engineering, Software Architecture for Cloud Environment.

Dr.S.Vimal, working as an Assistant Professor in the Department of Computational Intelligence, SRM Institute of Science and Technology Kattankulathur, has about 18 years of teaching experience. He received his BE degree in Computer Science and Engineering and M.Tech degree in Information Technology from Sathyabama University, Chennai. He received his Ph.D from Sathyabama Institute of Science and Technology. He has published more than 25 research papers in refereed International Journals and various international conferences. He has received several best paper awards for his research papers at various international conferences. His areas of research include Machine Learning, IoT, P2P Blockchain, Software Engineering. He is an active member of ACM and ICSES.