Nanotechnology Perceptions
ISSN 1660-6795
www.nano-ntp.com

Nutrition Deficiency Classification in Maize Plant Using Deep Learning Algorithms

Ashish Patel, Richa Mishra

Department of Computer Science and Engineering, Parul University, Vadodara, Gujarat, India
Email: ashish.patel28275@paruluniversity.ac.in

Maize, a vital global crop, is susceptible to various nutrition deficiencies that hinder its growth and yield. Identifying these deficiencies promptly is crucial for enhancing maize production. This research proposes a comprehensive approach for classifying nutrient deficiencies in maize plants using images of their leaves. Our methodology incorporates deep learning techniques. The four nutrient deficiency categories studied are Nitrogen (N), Phosphorus (P), potassium (K), and Zinc (Zn). In the deep learning approach, we harness the power of Convolutional Neural Networks (CNN), the pre-trained models Vgg19, Inception, and a hybrid CNN-SVM. Our experimental results show the effectiveness of both approaches in classifying nutrient deficiencies in maize plants. The deep learning approach, specifically the CNN-SVM architecture, exhibits significant promise, achieving an overall classification accuracy of 98% on the test set. This research underscores the potential of deep learning algorithms in facilitating plant nutrition management and, consequently, elevating crop productivity. The ability to rapidly and accurately detect nutrient deficiencies in maize plants using automated image analysis can be a valuable tool for farmers and agricultural experts, ultimately contributing to global food security and sustainability.

Keywords: Nutrient deficiency, Maize plant, Deep learning, Image classification, Agriculture.

1. Introduction

Technology is vital in every aspect of the burgeoning planet, and agriculture is vital for the sustenance of the human species. We continue to utilize conventional farming methods in our agricultural efforts. Agriculturalists still need help identifying nutritional deficiencies in their crops, which requires additional time, labor, and financial resources [1]. If a diagnosis is correct, there is a tendency for time, money, and productivity to be used. Typically, skilled specialists and agricultural laboratories identify areas for improvement in farmers. Various environmental variables can impair the accuracy of manual projections of nutritional deficits. The leaf is employed to detect nutritional deficiencies in crops, as these deficiencies can also appear in the stem, flowers, fruits, and other parts of the plant. A plant typically necessitates around twelve important nutrients to guarantee proper growth. The leaf is utilized to detect nutritional deficiencies in crops. A plant typically necessitates around twelve important nutrients to guarantee proper growth. Plants' primary nutrients require Nitrogen, phosphorus, potassium, magnesium, calcium, iron, molybdenum, sulfur, and chloride [2].

Maize, deductively known as Zea mays, is a profoundly huge harvest worldwide. It is an essential component of animal feed and a fundamental food source. With the rising worldwide populace and the interest in food security, upgrading maize production is basic. A significant factor in the cultivation of maize is the presence of nutritional deficiencies in the soil, which is significantly influenced by several factors. Inadequate measures of basic enhancements, similar to Nitrogen (N), phosphorus (P), potassium (K), and Zinc (Zn), can gigantically upset the turn of events and effectiveness of maize crops. These can be recognized by searching for various normal physiological signs in the leaves. Assuming we are to rapidly accomplish reasonable agribusiness, tending to these shortcomings is fundamental [3].

A plant's ability to intake water, nutrients, and sunlight is fundamental to survival. Nutrients accumulate in plants' organized structures. While plants contain macronutrients and micronutrients, they are more abundant because they are essential to a plant's growth and development. The macronutrients are (in order of abundance, with the two most abundant first) nitrogen (N) and phosphorus (P), followed by potassium (K), calcium (Ca), sulfur (S), magnesium (Mg), iron (Fe), zinc (Zn), copper (Cu), and manganese (Mn).

Lack of these macronutrients hinders the development of the leaves, hence reducing the production of simple foods. They attributed many developmental issues that affect plants, such as stunted growth, restricted arrangement of flowers, and even reduced generation of other natural products, to inadequate food. The growth rate of vegetation increases, the formation of new branches, and the change in the color of the leaves indicates nutrient deficiency [4]. The following table presents a complete analysis of the effects observed on the maize plant due to the destitution of micronutrients. Macronutrient deficiency symptoms

TABLE I. EFFECT OF MICRONUTRIENTS ON MAIZE PLANT

Macronutrients	Symptoms
Nitrogen (N)	Light green of upper leaves and yellow of lower leaves
Potassium (K)	Yellow and purple leaves with brown at leaves edge and poor flower and fruit.
Phosphorus (P)	Slow growth and yellow foliage
Magnesium (Mg)	Yellow between the leaf veins with red-brown tints and early leaves fall.

Entomologists have used mere observation to determine whether a nutrient-poor maize plant infects the plant in question. However, this methodology is very specific and demands much attention, as well as high professional skills and constant manual work, which make it vulnerable to the human factor. The increasing demand for food necessitates the progression of more efficient and accurate methods of diagnosing malnutrition in the plants of the maize crop. It is established in this case that there is a huge potential to enhance and optimize nutrient deficiency identification using Machine Learning and Deep Learning methods in leaves.

In this work, the key goal emerged to determine the primary nutritional deficits of the maize plants with the help of deep learning. Utilizing superior picture processing, this paper proposes to identify the specific nutritional value of the portion of the maize plant as seen in the leaf. The elements deficient in the maize plants in the collection of photographs are Zinc, potassium, Nitrogen, and phosphorus. This work aims to develop a feasible and repetitive method for identifying malnutrition facilitated by assessing geometrical characteristics. Enhanced learning approaches have been incorporated into the agriculture business. Year, there was an addition to this increasing field by evaluating how several significant learning computations affect the ability to understand the stimulating needs of the maize plants.

2. LITERATURE SURVEY

Artificial intelligence (AI) technology has recently entered plant pathology, allowing experts to identify and accurately classify pests and plant abnormalities. These innovations could change how plant diseases are distinguished, diagnosed, and controlled. This section will analyze a few "intelligent" (i.e., smart) systems that detect and discern plant pests and irregularities. The pros and cons of these systems will be discussed, as well as what the outcomes might mean for the future of their use in the discipline of plant pathology. Plant pathologists and agronomists have access to smart technology to help identify and classify plant ailments. This strategy will look into several artificial intelligence (AI) methods. These methods could yield important insights for detecting and identifying pests and plant defects. This article will survey the upsides and prerequisites of these innovations and their impact on plant pathology. The plant pathology disciplines commonly apply artificial intelligence techniques, with machine learning being the most prevalent. Two prevalent ML methods are the C4.5 classifier and a tree-based method. Both methods have been quite effective when plant diseases have to be classified according to their symptoms. These methods and calculations are also being applied to determine specific examples and side effects of nitrogen deficiency, making them appropriate for classifying diseases in their earliest stages.

Nevertheless, machine learning methods demand considerable labeled training data for previous plant pathology recordings. Irrefutably, a primary intelligence technique utilized in plant pathology is computer-based. Different machine-learning algorithms have been used to classify plant diseases. The c4.5 classifier, tree bagger, and linear support vector machines are among these algorithms. Each has computational features that make it a good candidate for diagnosing plant diseases, and each has at some point been heralded as a "best bet" for this particular application.

Nonetheless, employing them effectively demands a chunk of elbow grease in preparatory work that mostly has to happen in the past. The possibility of computer vision methods to successfully recognize plant disease is introduced by Singh et al. [5]. The PlantDoc dataset was utilized in this approach, specifically designed to distinguish and characterize plant diseases. The dataset comprises 3,451 data points that furnish details about 12 different species of plants and 14 different categories of diseases. The information in the dataset was obtained through a combination of web scraping and human annotation, which took 352 hours of labor to produce. Three models were trained to classify plant illnesses, showcasing the dataset's efficiency. The findings demonstrated a notable enhancement in accuracy, reaching a maximum improvement of 29%. The authors claim that this dataset has significant use in using computer vision algorithms to detect plant diseases.

Zhang et al. [6] introduced a novel approach to detect. For this purpose, an enhanced YOLOv3 algorithm that includes spatial pyramid pooling was used. The technology's centerpiece, deconvolution, combines oversampling and convolution to resolve accuracy problems. In this case, postures and sizes, which vary in each class of insects, cause many of these harmful pests to go undetected. A precision improvement was achieved by identifying a few more individuals per image. Then, averaged the results from 20 samples taken in natural locations for the pests' detection, leading to an identification rate of about 88%.

A method using fully convolutional neural networks was proposed by Wang and Zhang [7] for segmentation of the corn leaf disease. The process they propose includes several steps, beginning with acquiring image data. Then, following the acquisition of the image data, there is a step that equals visual enhancement. This enhancement facilitates the visual quality of the image data, and so, in turn, it aids the generation of the deep learning framework, that is, the segmentation test set and the training set. Once the image data are up to snuff, they are fed into a fully convolutional network. After the feature map is produced, a pair of deconvolutional (or up-sampling) steps are used to bring the segmented image back to the resolution of the original image.

The enhanced fully convolutional network (FCN) proposed in this work represents a significant step forward in the automatic diagnosing of diseases present at the country core level.[8] This work must first be acknowledged for its lineage to Wang et al. They are the ones who first proposed the automatic trait-extraction method. By characterizing healthy vs. diseased (lesion) leaves, we attempted to associate this work with the first three steps in the automatic diagnosing process: image capturing and processing. They focused on the second step, robust feature extraction from the FCN. Finally, we tried to properly evaluate the third step, the detection of disease symptoms, using precision (P), recall (R), and intersection over union (IoU).

The U-Net structure is highly favored for CNNs designed for image segmentation tasks. The architecture gets its name from its U-shaped figure, which connects the bottleneck to both network ends [9]. The feature extraction side (the encoder) consists of two convolutional layers and a down-sampling layer in each of the first four levels. The deepest level has the most features and the largest element size; the bottleneck has the least amount of activity and the most compressed representation of the input image. The decoder, in contrast, has four levels, working from the bottom to the top. Each of the first three levels on the decoder side doubles the output size of the previous level. Thus, the process also runs upward through the levels to "up-sample" the coarse representation of the prior level into an output map of the same scale as the target feature map for that level of the encoder. Maintaining a minimal number of visual discrepancies between the input and the output is precisely what the algorithm is supposed to accomplish: It is an image segmentation algorithm, after all.

Lin and others [10] Give a semantic division model that definitively recognizes and separates fine buildup at the pixel level in photos of cucumber leaves utilizing Convolutional Brain Organizations (CNNs). The proposed model scores 97.12 percent on typical pixel accuracy, 79.54 percent for joint intermingling extent, and 81.54 percent for dice precision when looking at a dataset with 20 models. The outcomes demonstrate that the proposed model performs better than laid-out division approaches, such as the Gaussian blend model, arbitrary woods, and fluffy c means. Even at a microscopic level, down to the resolution of each pixel, the proposed model achieves a surprising level of precision in identifying powdery mildew on cucumber leaves. Cucumber breeders need this ability to determine the extent of powdery mildew infestation.

Kerkech et al. [11] introduced a new method for identifying vineyard mildew by employing deep learning segmentation on photos captured by Unmanned Aerial Vehicles (UAVs). Utilizing a novel image registration technique to merge visible and infrared pictures captured by distinct sensors. This method ensures that the data from the two sensors are precisely aligned and integrated. The classification of each pixel into one of the specified classes—healthy, symptom, shadow, or ground—is then carried out using a fully convolutional neural network. The proposed method achieved impressive detection accuracy comparing levels of vines and leaves: 89% for vines and 84% for leaves. This shows that our work can be used to provide computer-assisted diagnosis of illnesses in vineyards.

Wang et al. [12] proposed the utilization of convolutional brain organization (CNN) models to analyze plant infections. The specialists used an item discovery framework incorporating VGG16, ResNet50, MobileNet, ResNet101, Quick R-CNN, and Cover R-CNN. The Fast Region-based Convolutional Neural Network (R-CCN) was utilized for disease detection. Interestingly, the Veil Locale-based Convolutional Brain Organization (R-CNN) definitively illustrated the impacted region. Different annotation techniques that involved object detection were used in the study. Fast R-CNN utilized labeling for the same purpose, whereas Mask R-CNN utilized Label Me. ResNet101 required the most time for training and testing due to its higher detection accuracy. MobileNet exhibited the quickest handling time yet forfeited accuracy, unlike ResNet101.

Verma et al. [13] evaluated three alternative convolutional neural network (CNN) models to determine how well they could assess the severity of late blight disease in tomato plants. These models, i.e., AlexNet, SqueezeNet, and InceptionV3, were applied to images of different types of tomato leaves taken at the PlantVillage collection. In this study, the authors used feature extraction and transfer learning to increase the predictive power of these models. They then fed the extracted features into a standard multi-class support vector machine (SVM). Finally, they assessed and compared the three models' efficiency based on the standard accuracy metrics, mean F1 score, and recall.

P. Wspanialy et al. [14] found that A specific residual neural network, the ResNet50 architecture, was used in the disease detection approach presented by Wspanialy et al. [10]. This approach depends on separating highlights from the foliage of tomato plants filled in nurseries and measuring the seriousness of the sickness by surveying the extent of the harmed leaf region. Adding normal qualities makes it conceivable to precisely anticipate results for new occasions in the assortment. The study also compared and contrasted the results of various alternative models, such as a system with ten categories (nine for various diseases and one for healthy leaves) and a binary classification method for separating healthy leaves from diseased ones. The models underwent training and testing using distinct sets of improved photographs. The study's findings demonstrated that a model comprising two distinct classes (healthy and unwell) had superior generalization compared to a model comprising ten classes.

Moreover, it was noted that ill leaves may be precisely distinguished just based on their morphology. Moreover, the proportional disease-damaged area is suitable for evaluating the intensity of distinct and restricted symptoms commonly induced by ailments like fungal and bacterial spots. Viruses can lead to specific symptoms in plants, including chlorosis (yellowing of leaves), leaf curl, and stunting. These symptoms are generally regarded as undesirable.

Renato et al. [15] developed a model based on a Convolutional Neural Network (CNN) and trained it using advanced transfer learning techniques. The model precisely identifies the nutrient deficiencies in banana plants; it is especially adept at identifying the deficiencies of Nitrogen, potassium, and sulphur. The dataset used included 995 photographs of banana leaves. We employed a "fine-tuning" technique using a pre-trained model called VGG 16. We also experimented with various input color spaces and different types of image preprocessing. The combined YUV color space and histogram equalization yielded the best results. As a whole, our work has an estimated accuracy of 98%.

In 2012, Jonilyn and her associates [16] developed an online mobile application that employs the Random Forest (RF) algorithm from machine learning to diagnose nutrient deficiencies in banana plants. The application's user follows a series of prompts that direct him or her to take specific photographs of the banana leaves in question. These images are then analyzed to determine whether the leaves are deficient in Nitrogen, potassium, or phosphorus. To achieve this, the application was trained using a 10-fold cross-validation technique, which yielded an overall accuracy of 92%. The training dataset comprises 705 photos, comprising 50 images of healthy leaves, 255 images of leaves exhibiting Nitrogen insufficiency, 155 images displaying phosphorus deficit, and 90 images of leaves manifesting potassium deficiency. Although extensive study has been conducted in the field of agricultural sciences, there is a requirement for accurate identification of micronutrient deficiencies in maize plants. Maize plants need to be addressed more in the study than other crops, leading to a need for more information. Comprehensive Our Databases store images of maize leaves exhibiting various nutrient deficiencies [17][18].

This study looks to fill this need by utilizing an unmistakable profound learning technique to precisely distinguish shortages of Nitrogen (N), phosphorus (P), potassium (K), and zinc in maize leaves. To effectively accomplish this goal, carefully gathering definitively marked photos that portray a different scope of supplement deficiencies is pivotal.

3. DEEP LEARNING ALGORITHMS

The proposed system employed a combination of CNN, InceptionV3, Vgg19, and CNN-SVM hybrid algorithms to classify and recognize nutrient deficiencies in Maize plants. This section provides a comprehensive explanation of each algorithm.

A. CNN

The architecture of the CNN algorithm for classifying the maize plant nutrient deficiency recognition is shown in Fig.1.

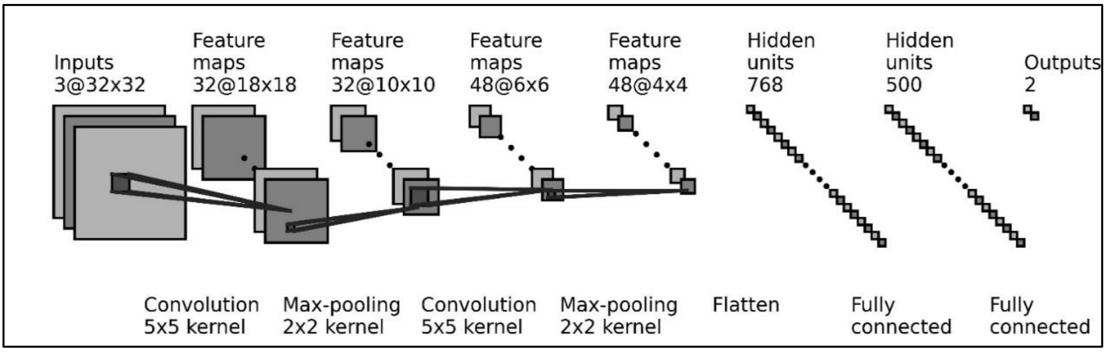


Fig. 1. Architecture diagram of the CNN algorithm

This convolutional neural network (CNN) is specifically optimized for picture classification tasks and performs them very efficiently. CNN is very good at extracting relevant features from images and composing those features to form a more complete representation of the input image. Its first layer has an input shape corresponding to the image's dimensions: height, width, and channels. After the first layer, the architecture has several convolutional layers, with two of them performing max pooling operations to decrease the dimensions of the previous layer's output while increasing the output's depth. The next-to-last layer performs a softmax operation to turn the last layer's output into a probability distribution over the classes the CNN can classify the image into [19][20][21].

The model applies a non-linear function called a "corrected straight unit" (ReLU). This mathematical operation takes a number; if it is positive, it just sends that number through. If it is negative, it sends a zero through. So, it has an effect like clipping negative values to zero, making it a good non-linear function for deep learning. After ReLUs are applied, the model applies a "max pooling" operation.

A max pooling operation takes a small, overlapping window. It slides it through the result of the convolutional layer (which we obtain before applying ReLUs) and looks for the maximum value.

Not long after, the capability of the ReLU implementation in the linear unit is used. It is used in the first convolutional layer. The model's generalizability ability increases when the number of dimensions (features) is reduced. It is not completely reduced. However, the output from that layer still has many features. To help out, a ReLU linear unit is then used in the first max pooling layer. The max pooling layer also helps the model to reduce the number of features (again, not completely, but by a considerable amount) and to increase the speed of the model's training time, which is a crucial thing for us since we intend to use the model repeatedly in the context of our research.

After applying the convolutional and pooling layers, the model smoothed the 2D element maps into a 1D vector. The data undergoes preprocessing to guarantee compatibility with the fully connected layers at this level. Leveling is a fundamental cycle that empowers the smooth change of the organization from the convolutional stages to the characterization stages. Sixty-four neurons with an activation function based on the Rectified Linear Unit (ReLU) comprise the first fully linked layer. This capability consolidates the acquired attributes from the past layers, permitting the model to get mind-boggling portrayals of the information.

To address overfitting, we use a dropout layer in our model. This layer randomly drops 20% of the neurons in the model during training. This makes the model more robust because it has to learn to approximate the function underlying the data without relying on memorization. The model's final layer has 6 neurons. Each of these neurons corresponds to one of the 6 classes in the classification problem. The softmax function generates a probability distribution across these 6 neurons. This facilitates the manufacturing of a product that is easily accessible. The result is easily understood, and the probabilities add up to 1 [22][23].

The CNN design compromises complexity and performance by using convolutional and pooling layers to extract information and then applying dense layers for multi-class classification.

B. InceptionV3

The architecture of the InceptionV3 algorithm for classifying the maize plant nutrient deficiency recognition is shown in Fig.2.

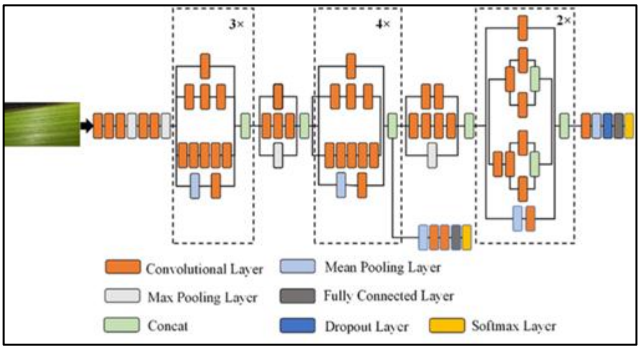


Fig. 2. Architecture diagram of the InceptionV3 algorithm

The InceptionV3 model is a highly reliable convolutional neural network architecture designed primarily for picture categorization tasks. The system is widely recognized for its efficiency and capacity to accomplish tasks. It employs different filter sizes and pooling operations in its inception modules to gather various properties from input photographs. The model is initialized with an input shape of (299, 299, 3), which is appropriate for RGB images with dimensions of 299x299 pixels. The model excludes the final layers responsible for categorization by setting the parameter include_top to False. This makes it possible to customize the output to meet specific requirements.

The architecture starts with the result tensor acquired by the InceptionV3 base model, which then, at that point, goes about as the contribution for the ensuing layers. The motivation behind a Worldwide Normal Pooling layer is to diminish the spatial components of the element maps, bringing about a compacted yield addressed by a solitary vector. The spatial hierarchy of the data is not altered during this compression process. The pooling approach reduces overfitting and the number of boundaries in the following layers. Two dense layers follow the pooling layer in the network, the first of which has 1024 neurons and the second of which has 512 neurons. In every dense layer, we utilize the activation function known as the rectified linear unit (ReLU) to introduce non-linearity into our model. This is of utmost importance because we want our neural network to approximate a function that is not just piecewise linear but also intricate and complicated in form. Once the dense layers have been added, a dropout layer is attached with a dropout rate 0.2. This layer makes overfitting less of a threat by turning off 20% of the neurons during training, but it does this in a stochastic way, that is, in a manner that acts like random noise. It turns off a different 20% of neurons in each training epoch. The noise it generates helps the network find a broader representation of the training set.

The final layer of the result consists of a thick slab of 6 neurons, corresponding to the number of classes in the grouping task. The softmax initiation capability here is a big help. It achieves a probability distribution over the six classes so that one or another makes for an easy decision, and it handles multiple classes more easily than any other method I know of. It is nice that the outputs can be interpreted as probabilities that sum to 1.

A new instance of the model is created by taking the base model's output and adding the specified categorization layers to it. Significantly, the layer settings are adjusted to immobilize ALL the layers of the original InceptionV3 model. This means that the 'trainable' parameter is set to FALSE for the layers of the base model. Thus, we use the pre-trained weights acquired from ImageNet while training only the newly added layers for the specific task.

The model uses categorical cross-entropy for the loss function. This function performs very well when having a problem categorizing data into many different classes.

The learning rate for the Adam optimizer is 0.0001. As a performance metric, the model is programmed to monitor accuracy. Beginning the model's activation. A comprehensive overview of the architecture as a whole is provided by the summary() function, which includes details about the output forms of each layer and the total number of trainable parameters. This makes it easier to comprehend the model's organization and complexity.

C. Vgg19

The architecture of the Vgg19 algorithm for the classification of the maize plant nutrient deficiency recognition is shown in Fig.4.

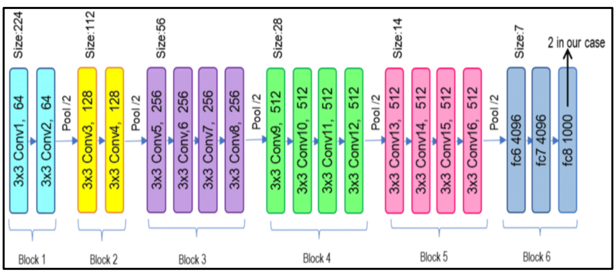


Fig. 3. Architecture diagram of the Vgg19 algorithm

The VGG19 architecture, a simple yet remarkably effective deep convolutional neural network for image categorization, has demonstrated its power across several computer vision tasks. It gets its name because it has 19 total weight layers, 16 convolutional layers, and three fully connected layers. This capability introduces the VGG19 model using pre-prepared loads from the ImageNet dataset. The contention include_top has a worth of Bogus. This disposes of the greatest classification layers and considers adaptability for explicit jobs. The input form is (200, 200, 3), indicating that images with RGB color channels and dimensions of 200 x 200 pixels are acceptable.

The names of layers and the objects of those layers are linked through the use of a layer dictionary. It considers the use of the highlights gathered by the VGG19 model. The resulting tensor is acquired from the block2_pool layer, a moderate portrayal of the information pictures, catching significant attributes while keeping up with spatial data.

The current VGG19 base has been innovatively expanded using a new supplementary layer design. The base now has an initial additional convolutional layer containing 256 filters, each with a kernel of size (2X2), that are all activated by the Rectified Linear Unit (ReLU) function. A max-pooling layer is next in line, and its use decreases the dimensions of the feature maps for this new base. The next layer is another convolutional layer containing 128 filters, which uses a (2X2) kernel that, once again, is activated by the ReLU function. Once again, a max pooling layer follows; its inclusion decreases the size of the feature maps even more.

The architecture progresses to fully connected layers by transforming the output from the preceding pooling layer into a flattened format. The compressed output is then sent into a densely linked layer of 64 neurons. The activation of this layer is achieved using the Rectified Linear Unit (ReLU) function, creating complicated representations. The model has a dropout layer with a dropout speed of 0.3 to determine the overfitting issue. During training, 30% of the neurons are deactivated randomly by the dropout layer. The dense layer of six neurons in the output layer represents the number of classes in the classification task. Over many classes, the softmax initiation capability creates a likelihood of circulation.

A Keras model occurrence is made by consolidating the contribution from the VGG19 model with the extra order layers, bringing about an extensive model. By altering the layer settings, the first five layers of the model are immobilized to improve the efficiency of the training process. The variable "teachable" is set to Misleading. This ensures that the pre-prepared loads of the primary layers stay unaltered during the preparation cycle, empowering the model to use gained qualities while changing the upper layers to the specific dataset.

The categorical cross-entropy loss function of the model makes it suitable for applications that require classification across multiple classes. The exhibition metric is precision, and the Adam enhancer is used with a learning rate 0.0001. The model's outline() capability gives a far-reaching outline of the model's engineering, including data about the all-out boundary count and the result types of each layer. This enables a broad cognizance of the model's plan and multi-layered nature.

D. CNN-SVM

The architecture of the CNN-SVM algorithm for classifying the maize plant nutrient deficiency recognition is shown in Fig.3.

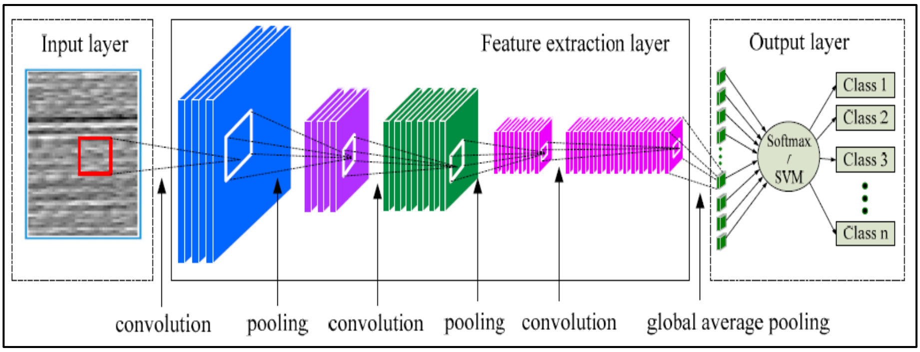


Fig. 4. Architecture diagram of the CNN-SVM algorithm

The CNN-SVM approach combines the strengths of Convolutional Neural Networks (CNNs) in feature extraction and Support Vector Machines (SVMs) in classification. As a result, this results in a strong hybrid framework that is highly suitable for problems related to picture categorization. The CNN component, which consists of numerous convolutional layers followed by pooling layers, is constructed using this strategy's sequential model. This strategy makes the purposeful extraction and upgrade of attributes from the info photographs conceivable.

The architecture begins with a convolutional layer that employs 16 filters on the input images with dimensions of 33, as specified by the input_shape. This layer is followed by the Corrected Straight Unit (ReLU) institution ability. This capability's consolidation of non-direct components enables the organization to learn complex examples. With a pool size of 22, the ensuing max pooling layer lessens the spatial components of the element maps, further developing interpretation invariance and bringing down computational costs.

Step by step, the model merges further convolutional layers, with an extension in the amount of directs in each layer: 32 in the resulting layer and 64 in the third layer. After each convolutional layer, the greatest pooling activity and a corrected direct unit (ReLU) enactment capability is performed. The pooling layer catches intricate designs and reduces the highlights' dimensionality.

After applying multiple convolutional and pooling layers, the model transforms the 2D feature into a 1D feature using a Flatten() layer. The interaction includes changing guides into a one-layered vector, permitting the information to be successfully used by completely associated layers. The model then incorporates 128 neurons in a fully connected layer. This layer is treated with the ReLU function, which enables the model to capture more complex data representations from the convolutional layers. To resolve the issue of overfitting, the model's engineering incorporates a dropout layer. A dropout rate of 0.2 has been set for the dropout layer, indicating that 20% of the neurons are disabled randomly during the training process.

The quantity of classes in the characterization task matches the thickness of the last result layer, which comprises six neurons. This layer uses the softmax initiation capability to create a likelihood circulation that envelops the different classes. To enhance the model's capacity to make precise predictions on novel and unfamiliar data and avoid overfitting the training data, we integrate L2 regularization. This regularization technique entails augmenting the loss function by incorporating a penalty term with a magnitude of 0.001. The penalty term serves to deter the model from allocating disproportionately high weights to its characteristics.

The CNN-SVM engineering utilizes profound figuring out removing highlights while guaranteeing hearty order execution with softmax result and regularization methods. The model's outline gives an exhaustive outline of its design, enumerating the result organizations of each layer and the all-out number of boundaries. This information is crucial to comprehending the model's complexity and capabilities.

4. PROPOSED METHODOLOGY

Figure 6 shows the block diagram of the deep learning system used to recognize nutrition deficiencies in Maize plants.

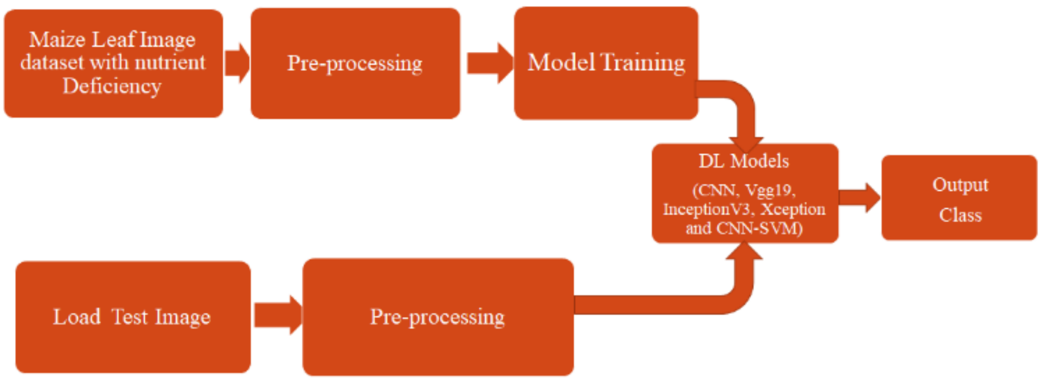


Fig. 5. Block diagram of the Maize plant nutrition deficiency recognition system

A. Maize Leaf Nutrition Image Database

The system collects images of maize leaves from a self-generated dataset to initiate the process. These photos showcase several types of nutrient deficiencies, such as Nitrogen (N), Phosphorus (P), Potassium (K), and Zinc (Zn) deficiencies. The photos serve as the basis for identifying and classifying nutritional deficiencies. The method used to create the dataset mentioned above focuses on detecting deficiencies and diseases in plants, particularly targeting four essential nutrients for maize: Nitrogen (N), Phosphorus (P), Potassium (K), and Zinc (Zn). The following is an in-depth discussion of each step involved in dataset development.

- Step 1: Identification of Key elements: The growth and output of maize plants depend on four important components: Nitrogen (N), Phosphorus (P), Potassium (K), and Zinc (Zn). These nutrients are recognized for their crucial function in the health and growth of plants.

- Step 2: Creating an artificial lack of nutrients. A total of 16 distinct treatment conditions are generated to replicate nutritional deficits. These treatments involve a mixture of insufficient nutrients and are used as experimental settings to examine the effects of nutrient limitation on plant growth.

- Step 3: Creation of Three Discrete Environments: The research is carried out in three different contexts: pot culture, field culture, and greenhouse culture. Each site is potentially distinct, with different soil, light, and climate conditions.
- Step 4: Photography Initiation: Photographic data are collected starting 20 days after the maize is planted. This allows us to analyze how plants respond to a lack of nutrients under various environmental conditions.
- Step 5: Gathering climatic condition data: Photo capture allows for monitoring meteorological factors. Each day, different climate-related attributes are recorded. These attributes include temperature, humidity, and other environmentally relevant metrics. The gathered data can help establish a relationship between plant responses and the conditions of the atmosphere surrounding the plants. Daily plant production count: The dataset consists of three habitats containing 16 treatments. Furthermore, there are two duplications for every treatment. Consequently, a total of 96 factories are undergoing daily inspections.
- The dataset production procedure involves capturing five shots of each plant regularly, resulting in a total of 480 images every day. The cumulative number of photos in the entire collection over 75 days is 36,000.
- The database has a total of 36,000 photos. This dataset provides a comprehensive resource for studying the effects of nutrient shortages on maize plants in different environmental situations. These photos are suitable for enhancing and evaluating machine learning and deep learning models for detecting plant nutritional deficiencies and diseases.
- Image sorting involves processing and saving photos from the dataset based on their resolution, camera angle, and backdrop.

The sample images of plants grown in pot and field are shown in Fig.2.

Fig. 6. Sample images grown for dataset creation in (a) Pot and (b) Field

The dataset distribution for the implementation of machine learning algorithms is presented in Table II.

TABLE II. DATASET DISTRIBUTION

Nutrition deficiencies	Total Number of Images	Number of Training Images	Number of Testing Images
ALL PRESENT	1470	1176	294
ALLAB	2430	1944	486
KAB	4301	3441	860

PAB	2970	2376	594
NAB	1535	1228	307
ZNAB	2545	2036	509
Total	15291	12201	3050

B. Image Preprocessing

Image preprocessing is utilized to improve the quality of photos before further processing and analysis. First, the provided photographs are utilized in RGB format. First, the RGB images are transformed into grayscale. The resulting images consist of a fair amount of noise. We used color transformation to deduce each image's specific hue and value. Finally, we used a median filter on each image to improve its quality, making it more lifelike and recognizable.

C. Classification using Deep Learning Algorithm

The proposed approach uses deep learning algorithms, such as CNN, InceptionV3, and VGG19, and a hybrid CNN-SVM algorithm to classify different nutrient deficiencies in maize plant leaves. We give the specifics of each algorithm in Section III. We fine-tune the hyperparameters of each algorithm to classify the images optimally and to achieve the highest classification accuracy. The convolutional neural network (CNN), a well-proven deep-learning algorithm in the image-processing domain, comprises a sequence of different types of layers, with the convolutional layers being the most important. A CNN will automatically and flexibly learn the spatial hierarchies of features from the input images. This study uses CNN as our basal method for categorizing three types of nutrient-deficient Chinese cabbage images and a nutrient-sufficient image.

InceptionV3 is a very well-known design for convolutional neural networks. It is very efficient and very accurate. A good reason for its being very well-known. It does not just use your kind of common-or-garden convolution; it also uses a lot of different sizes of convolutions, which allows it to pick up a lot of different kinds of visual patterns. You can think of it as being good at recognizing basic, more or less two-dimensional shapes. It is also good at more or less three-dimensional arrangements of those shapes, just recognizing many different kinds of visual or visual "stuff."

The model's ability to understand complex visual data makes it suitable for finding nutrient deficiencies in corn leaves. The VGG19 architecture has 19 layers and is a convolutional neural network (CNN) that has a simple design yet is quite effective in image classification tasks. It was pre-trained on a large image data set and used that "knowledge" to identify images when fine-tuned for specific tasks accurately. Despite how well it performs, it should be distinct from the best CNN for image classification because there are better CNNs.

The combined technique of CNN and SVM takes advantage of both worlds. The feature extractor is a CNN and SVM classifier. The CNN in this hybrid model is quite deep and can learn powerful features directly from the image pixels. The SVM is used to classify those powerful features into their respective nutritional deficiency classes. CNNs are state-of-the-art image classifiers. One would expect that using them to extract features from the images and then classifying them with an SVM, known for its classification accuracy, would yield excellent results.

For each algorithm to perform optimally, it must undergo a comprehensive and detailed hyperparameter adjustment process. Most people think of an algorithm as a single entity; however, one must remember that talking about a family of related algorithms within the overall structure of an artificial neural network is important. The specific parameters optimized for each method are outlined in Section III, showcasing their influence on the classification results.

D. Performance Evaluation

When evaluating our classification models, we must thoroughly analyze several factors such as precision, recall, F1 score, and accuracy. These features offer a thorough comprehension of the efficiency and accuracy of our trained algorithms in classifying brain MRI data.

- Precision: It is determined by dividing the count of true positive cases (accurately predicted positive cases) by the total count of positive cases predicted by the model. The model's capacity to minimize false positives is emphasized, rendering it especially valuable when false positives lead to significant expenses. Mathematically, precision is quantified by a certain calculation:

$$\text{Precision(P)} = \frac{\text{TP}}{\text{TP+FP}} \qquad (1)$$

- Recall: It, sometimes called sensitivity or true positive rate, measures the ratio of accurately identified true positive instances by the model to the overall number of positive cases. The model's proficiency in precisely capturing all pertinent positive examples is showcased. Mathematically, recall is formally defined as:

$$\text{Recall (R)} = \frac{\text{TP}}{\text{TP+FN}} \qquad (2)$$

- F1 Score: It is computed by taking the harmonic mean of precision and recall, serving as a performance metric for a model that attains a favorable equilibrium between the two. It considers both false positive and false negative outcomes and is particularly beneficial in cases with an imbalanced class distribution. The F1 score is computed using the following formula:

$$\text{Recall} = \frac{2\times\text{P}\times\text{R}}{\text{P+R}} \qquad (3)$$

- Accuracy: It is the ratio of correctly anticipated events to total events. The total number of instances in the dataset is determined by adding the number of true positives and negatives. Although precision is highly valued, it may not be the most suitable measure for addressing imbalanced datasets where one class is significantly more dominant.

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \qquad (4)$$

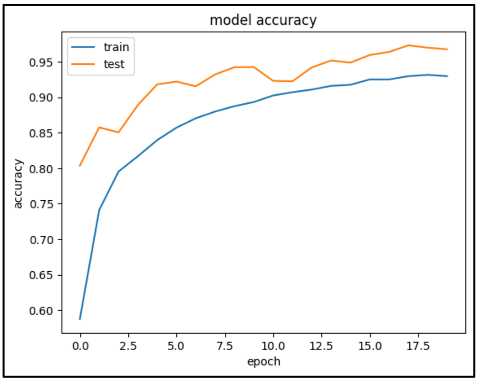
When confronted with classification issues, samples can be divided into categories referred to as True Positive (TP), False Positive (FP), True Negative, and False Negative (FN) accordingly.

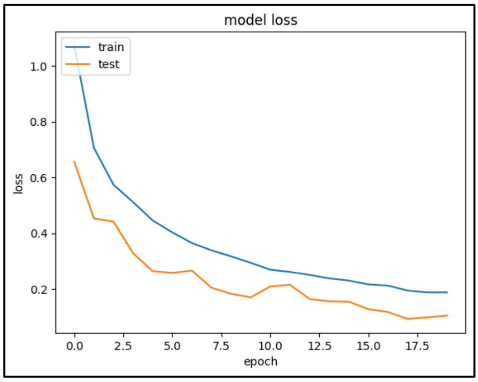
5. RESULTS

The outcomes of the suggested system are shown in this section. Each deep learning algorithm was trained using the Google Colab platform, taking advantage of its GPU. The performance of each algorithm was gauged using the precision, recall, F1 score, and accuracy metrics. The results of each algorithm are presented below in this section.

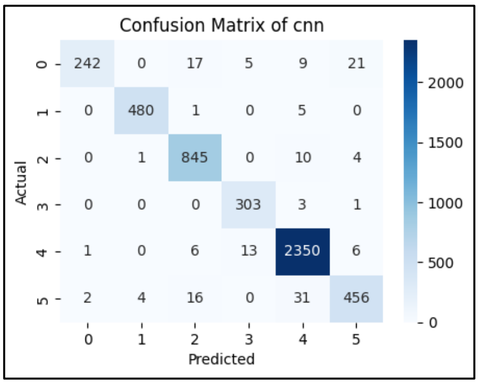
A. CNN

The performance of the CNN algorithm for maize plant leaf nutrition recognition is presented in Fig.8.





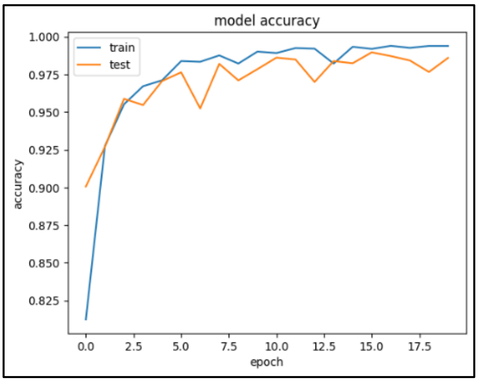
a b

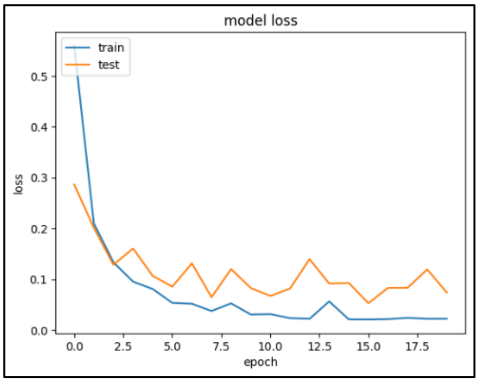


c

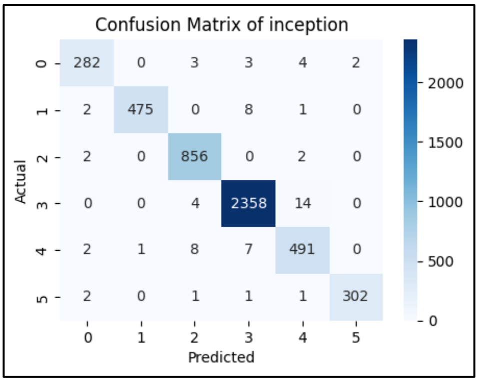
Fig. 7. The results of CNN algorithms for classification of maize plant leaf nutrient deficiency (a) Accuracy (b) Loss (c) Confusion Matrix

B. IncpetionV3





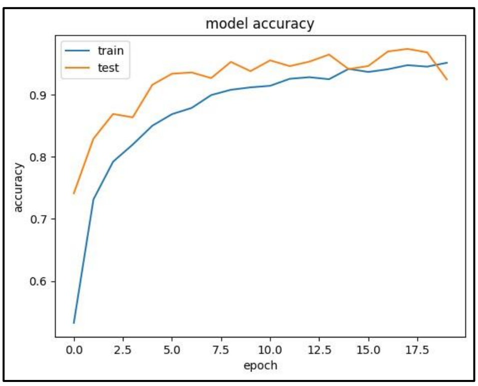
a b

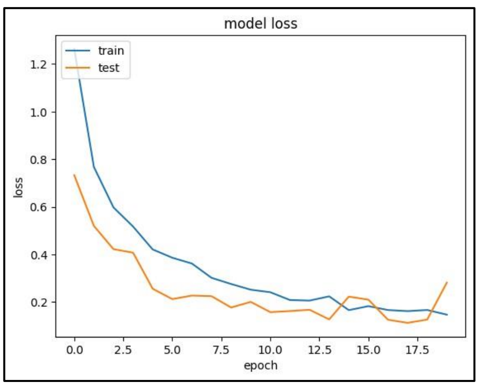


c

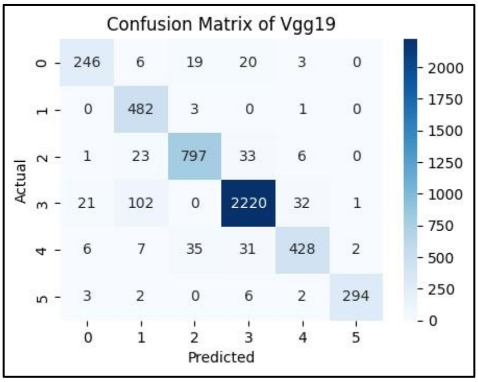
Fig. 8. The results of InceptionV3 algorithms for classification of maize plant leaf nutrient deficiency (a) Accuracy (b) Loss (c) Confusion Matrix

C. Vgg19





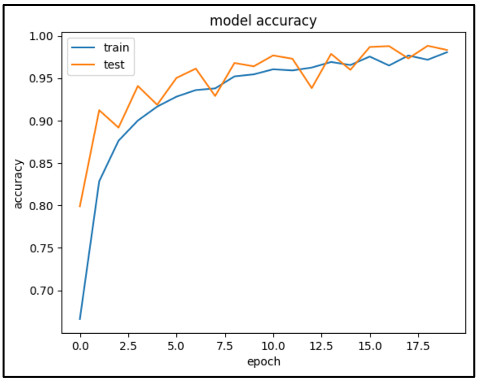
a b

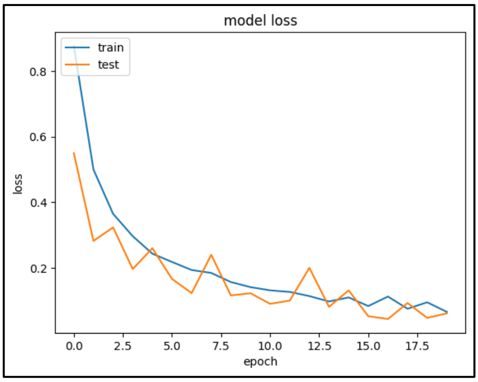


c

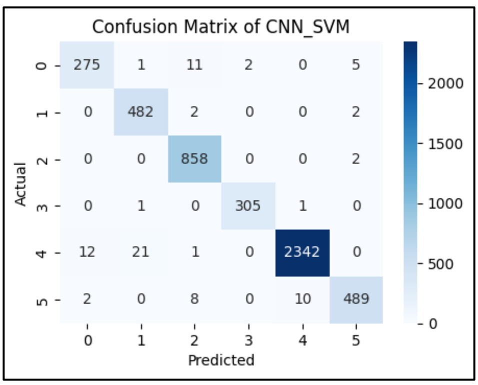
Fig. 9. The results of CNN algorithms for classification of maize plant leaf nutrient deficiency (a) Accuracy (b) Loss (c) Confusion Matrix

D. CNN-SVM





a b



c

Fig. 10. The results of CNN algorithms for classification of maize plant leaf nutrient deficiency (a) Accuracy (b) Loss (c) Confusion Matrix

Table III presents a comparative comparison of the performance of each algorithm in recognizing nutritional deficits in the maize plant leaf.

TABLE III. COMPARATIVE ANALYSIS OF DL ALGORITHMS FOR CLASSIFICATION OF MAIZE NUTRIENT DEFICIENCY

Algorithms	Precision	Recall	F1 Score	Accuracy	Training time
CNN	0.97	0.97	0.97	0.97	6208
InceptionV3	0.99	0.99	0.99	0.99	3604
Vgg19	0.93	0.92	0.93	0.92	8625
CNN-SVM	0.98	0.98	0.98	0.98	2564

An analysis compares the performance of deep learning algorithms when classifying nutrient deficiencies in maize plant leaves. Four models are examined: CNN, InceptionV3, VGG19, and CNN-SVM. The CNN model achieved very good results and was even the best in some aspects. However, it has our highest training time at 6208 seconds and a quite significant, comparatively speaking, training cost. The maize nutrient deficiency classification's precision, recall, and F1 scores are very close to the top scores achieved by InceptionV3. However, the accuracy of the CNN model is 0.97 compared to InceptionV3's 0.99.

On the other hand, the powerful model VGG19 yielded slightly lesser results. Performance metrics indicate that VGG19 had a precision of 0.93, recall of 0.92, F1 score of 0.93, and accuracy of 0.92. VGG19 also had the longest training time of the models analyzed, at 8625 seconds. This makes it inefficient compared to other models. The hybrid model of CNN and SVM, known as CNN-SVM, yields excellent results. Performance metrics indicate that CNN-SVM has a precision, recall, F1 score, and accuracy of 0.98. Moreover, its training time of 2564 seconds is less than that of any other model in our study. Although InceptionV3 is the most accurate model, CNN-SVM is a formidable contender when considering the balance of performance versus training time.

6. CONCLUSION

This study presents an extensive approach that uses advanced deep-learning algorithms to classify nutritional deficiencies in the leaves of maize plants. The system achieves accurate and consistent classification results using CNN, InceptionV3, Xception, VGG19, and a hybrid CNN-SVM approach. Each algorithm's performance is evaluated in detail using metrics including accuracy, precision, recall, and F1 score. Among the techniques analyzed, the hybrid CNN-SVM algorithm performed better. It achieved accuracy and exceeded precision, recall, and F1 score metrics. The CNN-SVM technique's hybrid nature offered a clear benefit: it required less training time than other deep learning models. As a result, its usefulness for practical applications increased. The suggested method reliably detects and treats nutrient shortages in maize plants by utilizing sophisticated deep-learning algorithms and optimizing their parameters. This technical advancement holds great promise for improving agricultural output and crop management techniques.

References

1. B. Kumar, V. ., N. P. Bajia, P. Chandra, and P. Dangi, "Title of the Paper," International Journal of Agriculture Extension and Social Development, vol. 7, no. 7, pp. 251-257, Jul. 2024. doi: 10.33545/26180723.2024.v7.i7d.801.
2. R. Duga, "Effect of Nitrogen, Phosphorus and Sulfur Nutrients on Growth and Yield Attributes of Bread Wheat," Journal of Ecology & Natural Resources, vol. 5, no. 1, 2021. DOI: 10.23880/jenr-16000230.
3. D. B. Fowler and R. H. Busch, "Crop Identification Using Cereal Science and Technology," in Handbook of Cereal Science and Technology, Revised and Expanded, 2nd ed. CRC Press, 2000, pp. 501-509. doi: 10.1201/9781420027228-49. Available: http://dx.doi.org/10.1201/9781420027228-49.
4. P. Thitisut, P. Sookwong, and S. Mahatheeranont, "Changes in contents of simple phenols and volatile compounds in leaves of black rice varieties following growth stages," Journal of Food Technology & Nutrition Sciences, vol. 1, no. 2, pp. 1-9, Dec. 2020. doi: 10.47363/jftns/2020(2)110.
5. Singh, D., Jain, N., Jain, P., Kayal, P., Kumawat, S., and Batra, N. (2020). "PlantDoc: A dataset for visual plant disease detection," in Proceedings of the 7th ACM IKDD CoDS and 25th COMAD. (Hyderabad, India: Association for Computing Machinery) 249–253.
6. Zhang, S., Huang, W., and Zhang, C. (2019). ScienceDirect three-channel convolutional neural networks for vegetable leaf disease recognition. Cogn. Syst. Res. 53, 31–41. doi: 10.1016/j.cogsys.2018.04.006
7. Wang, Z., and Zhang, S. (2018). Segmentation of corn leaf disease based on a fully convolution neural network. Acad. J. Comput. Inf Sci. 1, 9–18. doi: 10.25236/AJCIS.010002
8. Wang, Q., Qi, F., Sun, M., Qu, J., and Xue, J. (2019). Identification of tomato disease types and detection of infected areas based on deep convolutional neural networks and object detection techniques. Comput. Intell. Neurosci. 2019, 9142753. doi: 10.1155/2019/9142753
9. Shoaib, M., Hussain, T., Shah, B., Ullah, I., Shah, S. M., Ali, F., et al. (2022a). Deep learning-based segmentation and classification of leaf images for detection of tomato plant disease. Front. Plant Sci. 13. doi: 10.3389/fpls.2022.1031748
10. Lin, K., Gong, L., Huang, Y., Liu, C., and Pan, J. (2019). Deep learning-based segmentation and quantification of cucumber powdery mildew using convolutional neural network. Front. Plant Sci. 10. doi: 10.3389/fpls.2019.00155
11. Kerkech, M., Hafiane, A., and Canals, R. (2020). Vine disease detection in UAV multispectral images using optimized image registration and deep learning segmentation approach. Comput. Electron. Agric. 174, 105446. doi: 10.1016/j.compag.2020.105446
12. Q. Wang, F. Qi, M. Sun, J. Qu, and J. Xue, "Identification of tomato disease types and detection of infected areas based on deep convolutional neural networks and object detection techniques," Computational Intelligence and Neuroscience, vol. 2019, 15 pages, 2019.
13. S. Verma, A. Chug, and A. P. Singh, "Application of convolutional neural networks for evaluation of disease severity in tomato plant," Journal of Discrete Mathematical Sciences and Cryptography, vol. 23, no. 1, pp. 273–282, 2020.
14. P. Wspanialy and M. Moussa, "A detection and severity estimation system for generic diseases of tomato greenhouse plants," Computers and Electronics in Agriculture, vol. 178, p. 105701, 2020.
15. Guerrero R, Renteros B, Castaneda R, Villanueva A, Belupú I (2021) Detection of nutrient deficiencies in banana plants using deep learning. In: IEEE International Conference on

Automation

16. Dabalos JT, Gara GP (2017) LeafCheckIT: a banana leaf analyzer for identifying macronutrient efficiency, conference: the 3rd international conference on communication and information processing (ICCIP 2017)
17. Narayan, V., Srivastava, S., Mall, P. K., Kumar, V., & Awasthi, S. (2024). A theoretical analysis of simple retrieval engine. In Computational Intelligence in the Industry 4.0 (pp. 240-248). CRC Press.
18. Narayan, V., Srivastava, S., Faiz, M., Kumar, V., & Awasthi, S. (2024). A comparison between nonlinear mapping and high-resolution image. In Computational Intelligence in the Industry 4.0 (pp. 153-160). CRC Press.
19. Sawhney, R., Sharma, S., Srivastava, S., & Narayan, V. (2024). Ear Biometry: Protection Safeguarding Ear Acknowledgment Framework utilizing Transfer Learning in Industry 4.0. Journal of Electrical Systems, 20(3s), 1397-1412.
20. Gupta, A., Gupta, S., Mall, P. K., Srivastava, S., Saluja, A. S., Yadav, N., ... & Sriramulu, S. (2024). ML-CPC: A Pathway for Machine Learning Based Campus Placement Classification. Journal of Electrical Systems, 20(3s), 1453-1464.
21. Sawhney, Rahul, et al. "An Efficient Scientific Programming Technique for MRI Classification using Deep Residual Networks." Journal of Electrical Systems 20.2s (2024): 241-255.
22. Mall, P. K., Srivastava, S., PATEL, M. M., Kumar, A., Narayan, V., Kumar, S., ... & Singh, D. S. (2024). Optimizing Heart Attack Prediction Through OHE2LM: A Hybrid Modelling Strategy. Journal of Electrical Systems, 20(1).
23. Kumar, M., Kumar, A., Gupta, A., Srivastava, S., Narayan, V., Chauhan, A. S., & Srivastava, A. P. (2024). Self-attentive CNN+ BERT: An approach for analysis of sentiment on movie reviews using word embedding. Int J Intell Syst Appl Eng, 12, 612.