Journal of Computing & Biomedical Informatics
ISSN: 2710 - 1606

Volume 08 Issue 01
2024

Research Article
https://doi.org/10.56979/801/2024

Machine Learning-based Estimation of Soybean Growth

Samia Mishal[1], Hira Nazir[2*], and Muhammad Sami Ullah[3]

[1]MNS University of Agriculture, Multan, 64000, Pakistan.
[2]Department of Cyber Security, Emerson University, Multan, 64000, Pakistan.
[3]Govt. Graduate College of Commerce, Multan, 64000, Pakistan.
[*]Corresponding Author: Hira Nazir. Email: hira.nazir@eum.edu.pk

Received: August 19, 2024 **Accepted:** September 29, 2024

Abstract: The ability to determine the yield of soybean crop and the extent of the area of interest with non-destructive methods and at minimal cost is very important because of the increasing world population and the effects of climate change on crop productivity. Yield mapping is important in the crop field at the initial stage of crop management since it gives the total yield in the field, and the distribution of yield in the field which help in the decision making process of which area to fertilize, to irrigate, or to spray pesticide. Thus, the goal of this study was to improve the yield of the soybean crop and the return on investment while utilizing the yield prediction with image data with the least amount of resources and environmental pollution. The Problem Statement is conventional procedures, which are mostly based on the observation and statistical approaches, are often complex, time consuming and prone to errors. That is why the achievement of accurate and timely decisions for crop management is impossible with such restrictions The development of the presented modern technologies implies the search for new approaches that would improve the assessment of the growth with higher accuracy. The proposed research study makes use of Convolutional Neural Network (CNN) to predict the soybean crop yield from an RGB image dataset. The significant phases include data acquisition phase, data pre-processing phase, model-building phase, model-assessment phase, and model-testing phase. The layers in the CNN architecture include the convolution layer, pooling layer, activation layer and the dense layer as well as the output layer. The training of the model was carried in three steps and before each step, the hyperparameters of the model was adjusted. The model is proved to perform well and reliable for yield estimation. The accuracy of testing of the model, is 92.50% and validation accuracy is 94.59% whereas, the training accuracy is 100%.

Keywords: Machine Learning; Soybean Growth; CNN; Image Processing; Predictive Analytics; Agriculture.

1. Introduction

Soybean (Glycine max) is one of the most important and widely cultivated leguminous crops in the world. It has gained prominence due to its versatile uses, primarily as a source of protein, oil, and animal feed. Soybean is an annual plant that is grown for its seeds, which are rich in protein and oil content [1]. It is particularly valuable for its role in food production, livestock feed, and various industrial applications. Soybean has several key characteristics that make it a critical global crop, like nutritional value, oil production, Crop Rotation, Global Production. Like other crops, soybean cultivation faces challenges, such as pests, diseases, weather variability, and the need for efficient management to maximize yield. Therefore, accurate monitoring and prediction of soybean growth and yield are essential for sustainable production. The background of soybean underscores the need for advanced technologies, such as machine learning and remote sensing, to monitor and predict its growth and yield. These technologies can help address the challenges associated with soybean cultivation and contribute to increased agricultural productivity, sustainability, and food security.

The United States is a leader in agricultural research and has numerous institutions and universities involved in precision agriculture and machine learning applications in farming. Researchers in the U.S. have focused on a wide range of crops, including soybean, corn, wheat, and more. Australia, Brazil, China, Canada, and Europe Union countries are exploring machine learning and data-driven approaches to enhance crop monitoring and yield prediction.

Figure 1. Soybean

By incorporating GPs into agricultural decision-making processes, farmers can make more informed choices about crop management practices, leading to improved productivity, sustainability, and resilience in soybean production.

In order to utilize ensemble learning to boost soybean farming and surveillance from 50 farms using dataset from five seasons. The authors utilized the RF (Random Forest) and SVM (Support Vector Machine) and input features of SM and temperature, nutrient contents, and weather condition [2]. The integrated model considered more than 600 thousand pieces of information and achieved 88% recognition of soybean development stages. This high accuracy proved that hybrid machine learning models are capable of providing efficient growth monitoring solution to the farmers and therefore can guide the farmers to take necessary actions on the results to enhance the CNN (Convolutional Neural Networks) and LSTM (Long-Short Term Memory) networks are utilized by the authors in this research for the analysis of the growth of soybean. The data was a collection of satellite images and time series data of sixty farms for eight years in various growth phases and different environs. Hence, CNN-LSTM model was used to training for both spatial and temporal patterns and in terms of monitoring the growth, the model achieved 90% accuracy [3]. The authors conducted studies on the application of the compound models integrating k-means clustering and Artificial Neural Networks (ANNs) for the promotion of soybean growth. The data set contained field records and sensor data of 40 soybean fields over six consecutive years. Initially, K-means was used to categorize the growth patterns so that similar ones are grouped and then ANNs were used to make predictions of the growth stages [4]. This two-pronged strategy enabled the achievement and sustenance of about 85% accuracy in the monitoring of growth. The recommendations centered on increasing the application of unsupervised and supervised learning approaches in the enhancement of growth monitoring models for agriculture.

The GBMs (Gradient Boosting Machines) and Decision Trees in an ensemble learning to increase the chance of monitoring the growth of soybeans. The data was obtained from the field monitoring devices and questionnaires that were developed for 45 agricultural fields for ten years and growing seasons. The variables which were used when incorporating the ensemble model were the indicators of the health of the soil, the amount of rainfall and previous rates of production. The proposed model which is the combination of the favorable aspects of the GBMs and Decision Trees was able to perform well in the monitoring of the growth with a high accuracy of 87% [5]. From this work, it has been found that ensemble learning is very useful in capturing high interactions in agricultural datasets thus making it a reliable technique in enhancing the monitoring of growth and management of crops. An intelligent system is developed from a combination of GA and SVM with the objective of improving the observation of the growth rates of soybean. The information was obtained from thirty-five farms within seven cropping seasons including variables such as soil, climate, and other farmers' managerial variables [6]. Genetic Algorithms refined which input should be chosen for the features; the 94 features were then passed through the SVM model.

Regarding the assessment of growth phase identification, it was realized that the proposed hybrid model of the evolutionary algorithm and the SVM achieved a high of 86 percent, thus, highlighted the accuracy possibility of integrating evolutionary algorithms with SVM strategies in agricultural applications.

The authors used the following approaches: and Random Forest (RF) to improve the observation of the growth of the soybean. In particular, the data of 15 years of 60 soybean farms with various environmental conditions was analyzed. Bayesian networks worked on the probability of the random variables and RF algorithms on the other hand worked on the classification. In growth monitoring, the integrated model achieved 84 % accuracy, thus, fully capable of handling the fuzziness and the randomness inherent in agricultural data [1]. The results of this study were indicative of the potential of the further combination of probabilistic and ensemble methods toward aiding in decisions concerning the crop's management. In this study, the authors applied a MLP neural network with KNN for checking growth of soya beans. It was from over 20,100 data sets of 50 soybean fields from eight growing seasons with constantly changing conditions [7]. The software trained in MLP model was used for the growth stages in the help of input features like soil moisture, temperature, and vegetation indices and KNN was used for the optimization of predictions. The accuracy of the growth measurement integrated model was 83% which reveal the paradigm of incorporation of neural network with the instance based learning.

To achieve this objective, this paper aims at finding out whether is it possible to integrate fuzzy logic systems with CNNs for monitoring the growth of soybean or not. Data was collected from 45 soybean farms over nine years of growing seasons and information that was collected included the type of soil used, the weather conditions and the practices used in farming. The proposed research study makes use of Convolutional Neural Network (CNN) to predict the soybean crop yield from an RGB image dataset.

The images were obtained from Kaggle and each of these images was in the RGB format. First, the dataset could not be large enough for training CNN; thus, to enhance the dataset, image generator from TensorFlow was used, and the total images used were 7520. During the preprocessing, image resizing from 3456 x 4608 to 224 x 224, and rescaling were performed. The significant phases include data acquisition phase, data pre-processing phase, model-building phase, model-assessment phase, and model-testing phase. The layers in the CNN architecture include the convolution layer, pooling layer, activation layer and the dense layer as well as the output layer. The training phase has dropout to 0. 2, using Softmax and ReLU activation functions, applying early stopping, and optimizing with Adam optimizer at a 0. 0001 learning rate. Tools like accuracy, loss, precision, recall, confusion matrix, and F1 score confirmed the model's efficiency of soybean yield prediction. The training of the model was carried in three steps and before each step, the hyper parameters of the model was adjusted. The last model was proved to be perform well and reliable for yield estimation. According to the accuracy of testing of the model, it was 92.50% and validation accuracy is 94.59% while the training accuracy was 100%.

The main goal is to improve agricultural yield and production sustainability through the application of modern ML approaches. Therefore, when the growth of soybean is estimated well, the farmer will be in a better position to make the right decisions which will enhance on the yields and the use of resources. Another reason is to overcome the drawbacks of the conventional approaches to predict the crop growth where the interactions are complicated and cannot be easily modeled. This study also plans to use Remote Sensors (RS) data and apply advanced ML algorithms to give more accurate and reliable forecasts [8-9].

Using technologies like drones and satellite, the idea is to create technologies that deliver constant information about crops' condition and allow farmers to act promptly. The main objective of this study is to apply state-of-the-art ML methods to boost the efficiency and sustainability of agricultural production. Many of the practices in agriculture involve making observations and using simple statistical tools; these sort of approach are slow, bulky, and error-prone [10-12]. Thus, the application of ML along with CNN in estimating the probable development of the soybean will help the farmers in improving their decisions and efficiency of the resources.

The use of RS data including satellite and drone images and other improved ML algorithms to produce precise and trustworthy forecast. These techniques can work on large data, can find out the relations and can learn from new data; therefore, these are much superior to the earlier methods. It also explains the opportunities of the continuous control and management of crops in real time. Some of them are; Drone and satellite imaging as the world goes high tech these are used in capturing data on the health of the crops, the conditions of the soil and the environment [13-14]. Most of the applications that can update farmers on the status of crops so that they can deal with the changes. They can help in sensing pest attacks, diseases, and water stress at a stage that can be managed before the crops are affected and hence leading to enhanced production. Also, this research is aimed at increasing the efficiency of precision agriculture, which is the management of farming as a process that entails the provision of relevant information on farming. Therefore, the study supports the role of the application of ML in agriculture and the conversion of the traditional and unprofitable farming practices into new intelligent ones [15-20]. It is therefore an adaptation of climate change challenges and as a way of making a success of feeding the world in the future, through enhancing on the management of crops, the effects of climate change and efficiency of the inputs to be used. This shift is particularly important especially in relation to such global variables as population, climate and resource. Hence, by improving the reliability of the agriculture predictions, the ML can give a big boost to the efforts aimed at improving the sustainability of the farming [21-22]. Furthermore, this research aims at addressing some of the gaps that are currently witnessed in the existing literature as well as innovations with regard to implementation of technologies in the agricultural sector [23-30].

Thus, the objectives of this study include developing interfaces and decision support systems that incorporate easily interpretable ML models for the farmers and the agronomists. Data-driven farming is a feasible way to assist farmers in acquiring the right knowledge that can optimize their activities, combat climate change and feed the world's population. Finally, the research is focused on attaining the precision agriculture. Hence, this paper contributes to use the CNN-ML in agriculture for predicting soybean yield (high or low growth). Moreover, recall value, confusion matrix, and f1 score values are used for evaluating the performance of the model.

2. Materials and Methods

The subsequent sections of this paper detail the methods and procedures applied in this research:

Dataset: The source of the dataset, the type of dataset, and the total number of images has been explained.

Applied Preprocessing: The applied pre-processing techniques are discussed in this section for model training, and it also explains the importance of these techniques and their necessity to improve model predictions.

Model: It has been explained what and why a specified model has been chosen and applied to achieve the objective of this study work.

Training: The purpose of model training and its hyper-parameter tuning is explained in this section. It has also been discussed what are major factors that contribute towards the better development of this model.

Evaluating the performance: This is an important section that ensures the validity of a model. The model can be evaluated by means of accuracy achieved by the model, the loss of the model during training and validation phases, precision of the model for predicting soybean yield, recall value, confusion matrix for evaluating prediction results, and f1 score value. All these mentioned metrics are major scales to evaluate the performance of the model.

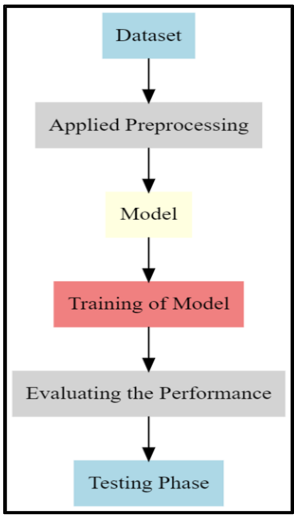


Figure 2. Proposed Methodology

2.1. Datasets

The dataset is retrieved from an online dataset repository (i.e., Kaggle) [31]. It consists of RGB images of soybean plant leaf images having two classes: high yield and low yield. The total number of images in high yield is 110, and the low yield consists of 266 images.

Table 1. Sample images from dataset

Low yielding crops of Soybean	High yielding crops of Soybean

2.2. Data Augmentation

These 370 images are not enough so we applied data augmentation technique on dataset for increasing images. After this technique the total number of images was increased to 7,520 these images were enough to train the model. Data Augmentation is a technique in which new images are formed on the basis of existing images. The image is given as input and new image is generated through different process such as rotation, cropping, flipping, resizing, brightness adjustment, shearing and saturation adjustment are commonly used for this process. TensorFlow has developed image generated library for this purpose. This library is used in this research for generating images through data augmentation.

Table 2. Number of images in the dataset

Images	Original	After Augmentation
Healthy	110	2200
Diseases	266	5320
Total	370	7520

2.3. Applied Preprocessing

It can be defined as a process that prepares the raw image data for use and to also comprehend them. This is a big step as it allows eradicating other unwanted distortions and, at the same time, increasing other significant aspects of computer vision. It improves the effectiveness of the image data for the deep learning algorithm by pre-processing the image data for the training process. In this phase of image preprocessing some of the following methods are used. A prerequisite to the training of a model is the processing of the data. It is one in which images are converted to a new shape or format that the model can recognize and takes less computational power to train.

The applied preprocessing is rescaling and resizing. It is shown in Figure 3, and 4.

Resizing is a process in which the height and width of image is set to a new value. In this study, the original size of input image was 3456x4608; this was resized to 224x224. The reason for this is; all the images are resized to the same dimension that leads the model to faster model development.

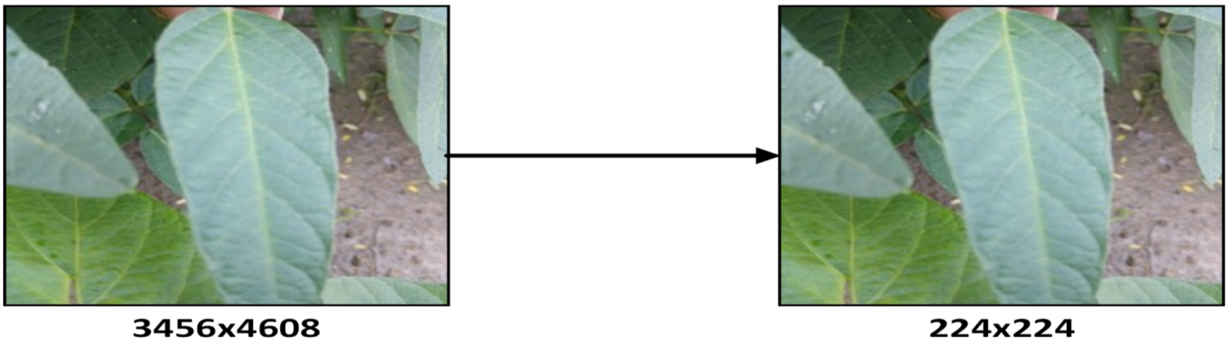


Figure 3. Resizing process of images

Also, square images are useful if one wishes to use prior models or layers in the CNN structures. Majority of the high performing CNN models are trained on images of a square shape. This is because by making our input images to be square shaped, it will be easier to use these pre-trained models since the dimensions will also match. This makes it possible to use transfer learning where the features learned by the pre-trained models can enhance the accuracy and rate of own CNN models.

For example, let there be a set of images of different dimensions such as 600X600, 1200X1200, 1000X500, etc. Using these images as inputs to the CNN model as they are would be an issue in defining the input layer and how to deal with several dimensions when training. But by resizing all the images to the same size of square for instance 224×224 pixels we make the dimension of all images to be the same hence making the model designing and training to be easier.

The second applied preprocessing step is rescaling. Rescaling is a process in which each pixel (originally, it is in the range of 0-255 for RGB images) is transformed to a new value in the range of 0-1. This is a necessary process that must be applied to images before training the model [32].

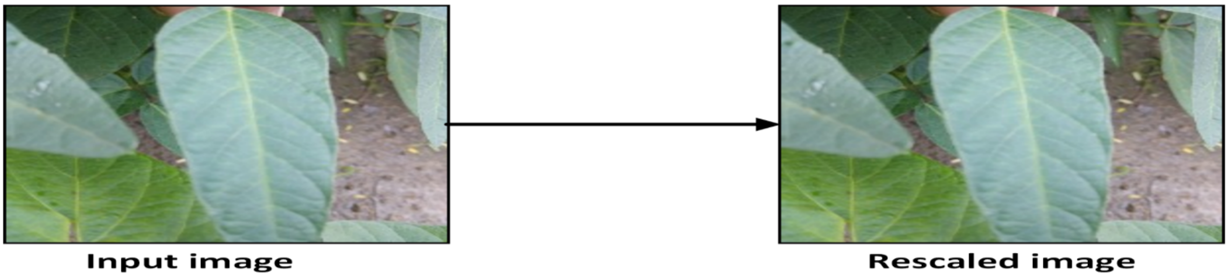


Figure 4. Rescaled process for input image

It also makes the computations more reasonable while at the same time it reduces the aspect of domination of large numbers. These preprocessing steps assist in reducing the amount of computational power required to train the model and the time taken to train the model was reduced to a few hours from days. Mathematically, normalization $f_{normalized}$ is expressed as following equation:

$$f_{normalized} = \frac{input\ Pixel\ Value}{255}$$

This rescaling process is also termed as image normalization.

2.4. Model

CNN is an advanced machine learning (ML) algorithm that has ability of automatically extracting features from images and then classifying those types of images [33-35]. As compared to traditional ML algorithms that were manual and required more time and feature extraction and selection process CNN outstands those algorithms. This is the reason of choosing CNN algorithm. In this study work a custom CNN model was defined with various stacks of layers including the following:

- Convolutional-layer
- Pooling
- Activation
- Dense or fully connected layer
- Output layer

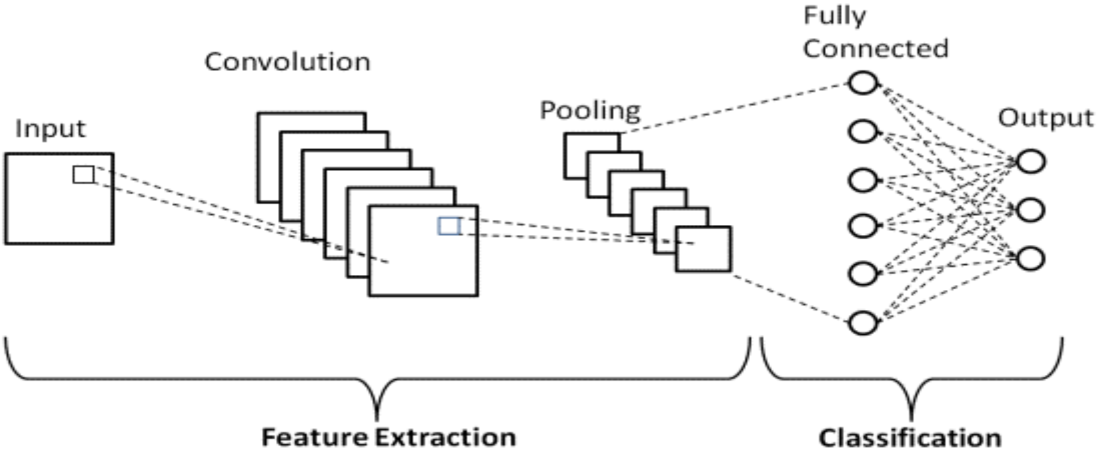


Figure 5. Basic architecture of CNN

Input Layer: input_shape=(224, 224, 3)

This layer determines the input for image shape. In this case, images are in the format of 224 * 224 and 3 bands of color (RGB).

First Convolutional Layer: tf. keras. layers. Conv2D(32, (3, 3), activation='relu')

This layer has 32 filters, of size 3 x 3. These filters are applied on the input image to extract features and ReLU activation function is used here to introduce non-linearity in the model.

First MaxPooling Layer: tf. keras. layers. MaxPooling2D((2, 2))

This layer applies max pooling with a filter of size 2×2. It decreases the spatial dimensions of the feature maps, which means that it subsamples the input, which in its turn is helpful in terms of lessening the computational complexity and the identification of the main features.

Second Convolutional Layer: tf. keras. layers. Conv2D(64, (3, 3), activation='relu')

The last layer of this network is composed of 64 filters of the size of 3 x 3. Similar to the first convolutional layer, it takes the output of the previous layer and extracts more complex features from it using ReLU activation function.

Second Max Pooling Layer: tf. keras. layers. MaxPooling2D((2, 2))

This layer again does 2x2 max pooling on the data obtained from the previous layer.It also diminishes the spatial dimensions one more step to point the network at the features of interest.

Third Convolutional Layer: tf. keras. layers. Conv2D(128, (3, 3), activation='relu')

This layer has 128 filters of size 3 x 3. It extracts even higher level and more abstract features from the input with the help of ReLU activation function.

Third MaxPooling Layer: tf. keras. layers. MaxPooling2D((2, 2))

This layer carries out another 2 by 2 max pooling operation on the image. It further decreases the spatial dimensions, which is helpful in transforming the extracted features into a more compact form.

Flatten Layer: tf. keras. layers. Flatten()

This layer is to flatten the 3D output from the previous layer into a 1D vector. This is important to move from the convolutional layer to the fully connected layer of the neural network.

First Fully Connected (Dense) Layer: tf. keras. layers. Dense(128, activation='relu')

This layer has 128 neurons and applies the ReLU activation function.

Output Layer: tf. keras. layers. Dense(num_classes, activation='softmax')

This layer has the number of neurons as the number of classes which is two in this study and is defined as num_classes in the set. In Multi-class classification, it generates the probability distribution of the classes with the help of softmax activation function.

Functionality and Parameters: Convolutional Layer: These layers are utilized for the feature extraction process, and last layer id for output. These layers process the input image of soybean leaves to extract the edge, texture and shape of the soybean leaves.

Activation Function (ReLU): The ReLU (Rectified Linear Unit) function is then used as activation function in the model as the model is able to capture simple patterns hence the need to introduce non-linearity in the model.

Pooling Layers (MaxPooling2D): These layers are useful in down sampling the feature maps and only the large feature of the feature maps is retained and to make the feature map smaller.

Flatten Layer: This layer is useful because it will have transformed the 3D feature maps into a 1D vector as fully connected layers only accepts 1D inputs. Two-dimensional Feature map generated by max pooling layer is converted into one-dimensional single array.

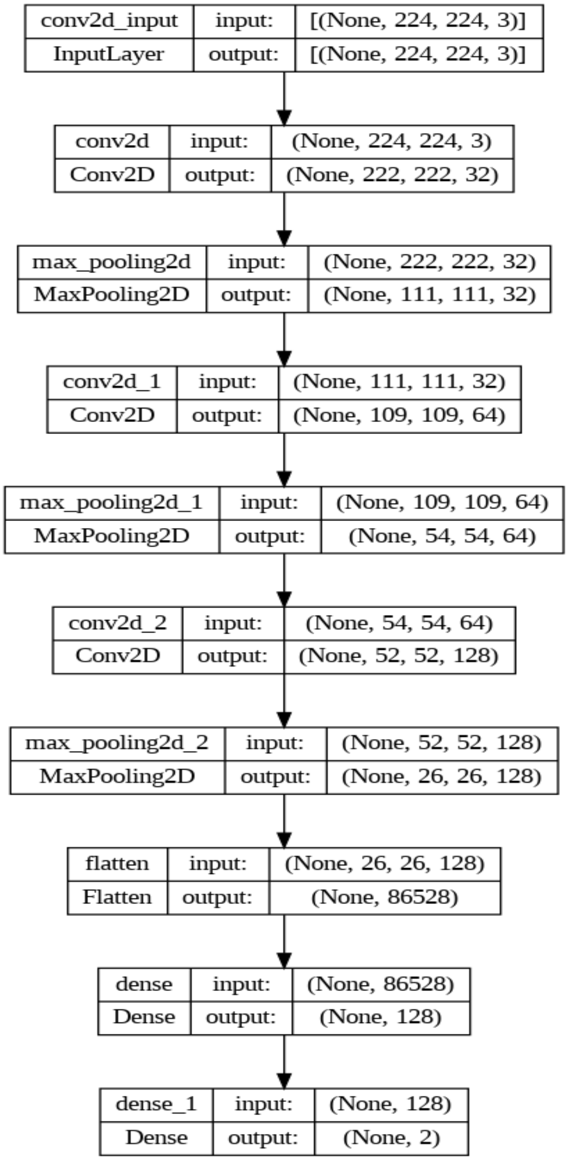

conv2d_input | input: | [(None, 224, 224, 3)]
InputLayer | output: | [(None, 224, 224, 3)]
conv2d | input: | (None, 224, 224, 3)
Conv2D | output: | (None, 222, 222, 32)
max_pooling2d | input: | (None, 222, 222, 32)
MaxPooling2D | output: | (None, 111, 111, 32)
conv2d_1 | input: | (None, 111, 111, 32)
Conv2D | output: | (None, 109, 109, 64)
max_pooling2d_1 | input: | (None, 109, 109, 64)
MaxPooling2D | output: | (None, 54, 54, 64)
conv2d_2 | input: | (None, 54, 54, 64)
Conv2D | output: | (None, 52, 52, 128)
max_pooling2d_2 | input: | (None, 52, 52, 128)
MaxPooling2D | output: | (None, 26, 26, 128)
flatten | input: | (None, 26, 26, 128)
Flatten | output: | (None, 86528)
dense | input: | (None, 86528)
Dense | output: | (None, 128)
dense_1 | input: | (None, 128)
Dense | output: | (None, 2)

Figure 6. Layered architecture of custom defined CNN model

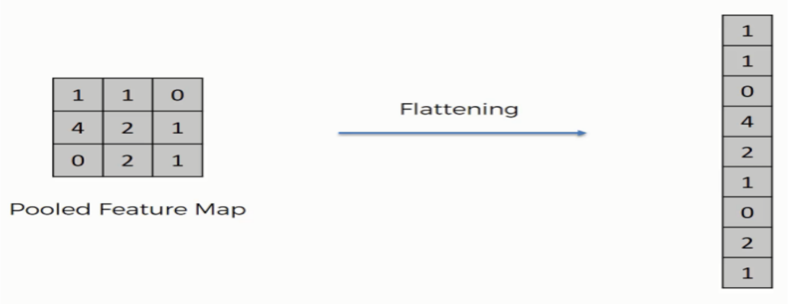


Figure 7. Working principle of flatten layer.

Dense Layers: These layers are the fully connected layers and here the higher order thinking process is performed through the features obtained from the convolutional layers. The first complete connection layer integrates all the features, while the last layer yields the probability of each class.

Softmax Activation: This function is utilized in output layer in multi class classification to provide the probability distribution of all the classes [36-37].

This CNN architecture is used to diagnose the crop yield by analyzing the images of leaves of soybean plants. It first extracts features using several convolutional layers and pooling layers followed by fully connected layers to make predictions based on these features. It incorporates pre-trained layers to increase the model's performance and precision; additionally, it constrains inputs to increase the computational speed and maintains the aspect ratio to prevent image distortion. All the layers with their Corresponding values are detailed in Table 3.

Table 3. Layered description of applied model

Layer (Type)	Output Shape	Param#
Conv2d	(222,222,32)	896
Max_pooling2d	(111,111,32)	0
Conv2d_1	(109,109,64)	18496
Max_pooling2d_1	(54,54,64)	0
Conv2d_2	(52,52,128)	73856
Max_pooling2d_2	(26,26,128)	0
Flatten (Flatten)	(86528)	0
Dense (Dense)	(128)	11075712
Dense_1 (Dense)	(2)	258
Total Parameters	11,169,218	(42.61MB)
Trainable parameters	11,169,218	(42.61MB)
Non-Trainable parameters	0	(0B)

The loss function is added to reduce the model's loss for soybean yield predictions using image data. In this work a categorical cross entropy function was employed as a loss function. The chosen loss function is highly suitable for CNN-based models. It is applied during the training phase for model learning. This function calculates the disparity between the predicted distribution and the actual distribution of each class. It adds the negative logarithm of the predicted probabilities to the corresponding classes. This loss value increases when there is a false prediction by model.

2.5. Model Training

In the next step, all these images were divided into train_set, validation_set, and testing_set with 70:20:10 percent ratio. The main objective of this division was to develop a better improved classification model that could have a good generalization ability for the soybean yield estimation.

First of all, the training stage is the stage at which the model that is going to develop is initiated. Here, it gets engaged in a particular training dataset and looks for the concealed patterns in input images of soybean leaf images and correlation extremely actively. Thus, in the context of the methods of supervised learning, the model searches for the specific input characteristics that will correspond to the given target outputs – the so-called labeled examples of soybean, for instance, healthy or diseased. Within each stage, the model modifies the parameters concerning the productiveness of the model in an effort to acquire the smallest training error.

Finally, after the training stage is complete, the model goes through the validation stage. Here, it uses the obtained knowledge to predict response for observations in another data set called the validation set. This dataset is used separately to test the performance of the model it is very beneficial in cases of tuning the model parameters. Slowly, adjust these hyper parameters with a lot of care to get the model with least possible error. This model is to be considered as the estimate of the test error rate on unseen data and this essentially means that it holds the potential of yielding the best results.

Following the validation, our model faces its ultimate test: The set for testing the proposed model of analytical relationships is the following: Since this is an independent dataset from the training and the validation dataset, this is a factor that would provide the best estimation of the final trained model.

The dropout value is set to 0.2 to avoid over-fit issues that might occur during the training phase of the model, activation functions used are Softmax and ReLU for probability estimation and for the activation of neurons of inner layers respectively [36]. To reduce the probability of over-fitting early stopping function was applied, Adam optimizer was used to train the model with less computational resources and time was also less, learning rate was set equal to 0.0001, while performing training the loss of the model was checked and regulated to the minimum by Cross-entropy function.

2.5.1. Evaluation Metrics

Accuracy, precision, loss function, recall, confusion matrix, and f1 score are employed in this study work to evaluate the performance of applied CNN model.

2.5.2. Confusion Matrix

The efficacy of model is assessed using a table called the confusion matrix. In this case, "true positive" model makes correct positive predictions. "False positive" are when the model wrongly predicts a positive label when the actual label is negative. "True negative" are when model makes correct negatives predictions.

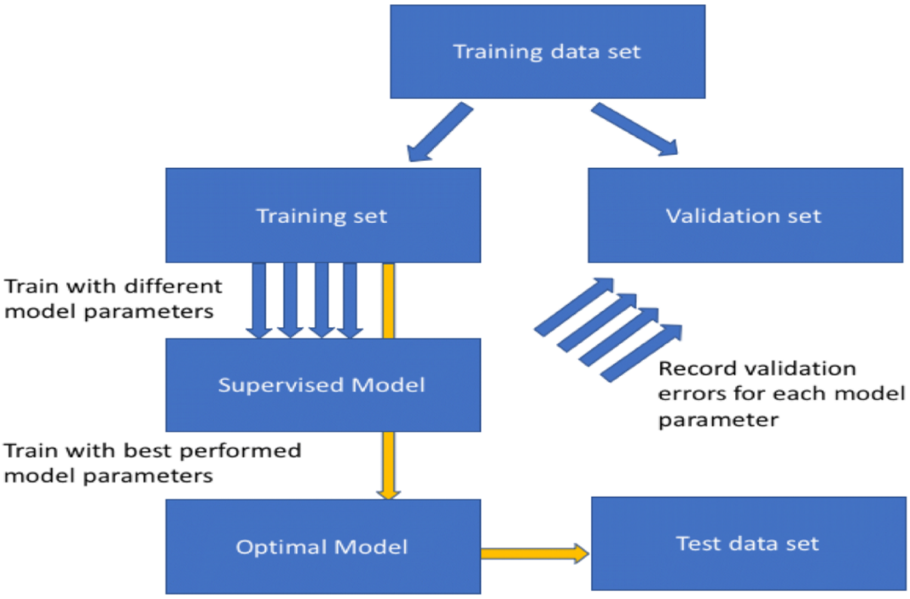
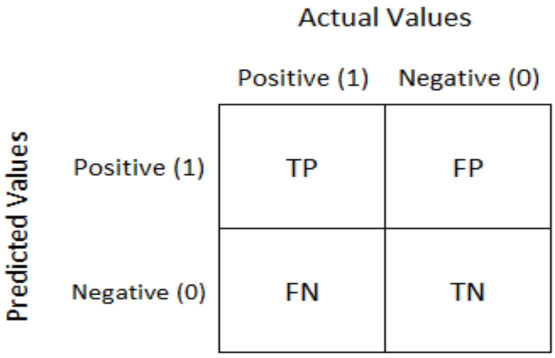

Training data set
Training set
Validation set
Train with different model parameters
Supervised Model
Record validation errors for each model parameter
Train with best performed model parameters
Optimal Model
Test data set

Figure 8. Training phase of model with dataset

Table 4. Hyper-parameters for 1st training phase of model

Parameter Name	Value/Status
Epoch size	50
Learning Rate	0.0001
Activation Function	Softmax
Dropout	0.2
Activation function for internal layers	ReLU
Optimizer	Adam
Early Stopping	Enabled
Loss function	Cross-entropy
Batch size	32



	Positive (1)	Negative (0)
Positive (1)	TP	FP
Negative (0)	FN	TN

Figure 9. Confusion matrix working principle, TP is true positive, FP is false positive, FN is false negative, and TN is true negative.

Accuracy: It is used before compilation of model. It is used to specify prediction of developed model after training phase. It is used to measure correct predictions of trained model. It is primary metrics to evaluate the performance of model. It is simple to implement and can be used to compare with performance of other models [38].

$$Accuracy = \frac{True\_positive + True\_negative}{True\_positive + True\_negative + False\_positive + false\_negative}$$

Recall: It calculates true positive predictions of soybean yield prediction over the total real positive samples given in testing set.

$$Recall = \frac{True\_positive}{True\_positive + False\_negative}$$

Precision: It calculates positive predictions of soybean yield over the total positive input samples [39].

$$Precision = \frac{True\_positive}{True\_positive + False\_positive}$$

F1 score: The mean value of precision and recall used to measure performance of model is termed as f1 score, as given [40-41]:

$$f1\ score = 2\ \frac{Precision * recall}{precision + reacll}$$

3. Results & Discussion

The training results of 1st training phase with each epoch are shown in Table 5. In this training phase the epoch size was set to 50 but training stopped after 14 cycles due to early stopping function, the batch size was 32, optimizer was adam, Softmax and ReLU were activation functions, learning rate was set to value of 0. 001, dropout value was set to 0. 2, and crossentropy was the function to measure loss of model during training time.

Table 5. Training results for 1st phase of model

Epoch	Loss	Accuracy	Val_Loss	Val_Accuracy
1/50	0.6099	0.7088	0.4706	0.7027
2/50	0.4652	0.7088	0.3407	0.7432
3/50	0.3779	0.7893	0.2921	0.9054
4/50	0.3779	0.8429	0.2732	0.9324
5/50	0.3420	0.8736	0.2537	0.9054
6/50	0.3247	0.8774	0.2674	0.8649
7/50	0.2533	0.9234	0.3177	0.8378
8/50	0.2300	0.9310	0.1895	0.9324
9/50	0.1975	0.9272	0.1728	0.9459
10/50	0.1638	0.9579	0.2956	0.8243
11/50	0.1716	0.9310	0.2125	0.9054
12/50	0.1619	0.9272	0.1570	0.9324
13/50	0.1092	0.9732	0.2138	0.9054
14/50	0.1071	0.9808	0.1408	0.9459

The other evaluation metrices such as precision, recall, f1-score, and support values are shown in Table 6.

Table 6. Evaluation metrices results for 1st training phase

	Precision	Recall	F1-score	Support
High Yield	0.83	0.95	0.89	440
Low Yield	0.98	0.92	0.95	1060
Accuracy			0.95	1500
Macro avg	0.90	0.93	0.92	1500
Weighted avg	0.93	0.93	0.93	1500

The precision is 96%, recall is 95%, and f1 is also 95% for this training phase of CNN. After this the model was again trained with hyper-parameters as used in previous training but this time the model was trained different learning rate. The hyper-parameters employed are shown in Table-7 along with their Coressponding values. This resulted in the accuracies and loss as shown in training results Table 8. Whereas, evaluation metrices with improved values such as precision, recall, f1-score, and support values are shown in Table 9.

Table 7. Hyper-parameters for second training phase of model

Parameter Name	Value/Status
Epoch size	50
Learning Rate	0.001
Activation Function	Softmax
Dropout	0.2
Activation function for internal layers	ReLU
Optimizer	Adam
Early Stopping	Enabled
Loss function	Cross-entropy
Batch size	32

Table 8. Training results per epoch for second training phase

Epoch	Loss	Accuracy	Val_Loss	Val_Accuracy
1/50	1.5317	0.5057	0.4878	0.7027
2/50	0.4876	0.7126	0.3140	0.8378
3/50	0.4027	0.7701	0.3008	0.8649
4/50	0.3690	0.8123	0.2593	0.8919
5/50	0.3046	0.8467	0.2901	0.8378
6/50	0.3790	0.8046	0.2702	0.8649
7/50	0.3769	0.7663	0.2983	0.8649
8/50	0.2520	0.9004	0.2690	0.8919
9/50	0.2974	0.8927	0.2249	0.8919

Table 9. Evaluation results for model training

	Precision	Recall	F1-score	Support
High Yield	0.80	0.92	0.85	440
Low Yield	0.96	0.90	0.93	1060
Accuracy			0.91	1500
Macro avg	0.88	0.91	0.89	1500
Weighted avg	0.91	0.91	0.91	1500

After this the model was again trained with hyper-parameters as used in previous training but this time the model was trained without early stopping function. The hyper-parameters employed are shown in Table 10 along with their Coressponding values.

Table 10. Hyper-parameter setting for model training

Parameter Name	Value/ Status
Epoch Size	50
Learning Rate	0.001
Activation Function	Softmax
Dropout	0.2
Activation function for internal layers	ReLu
Optimizer	Adam
Early Stopping	Disabled
Loss Function	Cross-Entropy
Batch Size	32

At this training with hyper-parameters mentioned in Table 10, the model resulted in the following training and evaluation results depicted in Table 11, and Table 12.

Table 11. Model training results

Epoch	Loss	Accuracy	Val Loss	Val Accuracy
1	0.6733	0.6475	0.5662	0.7027
2	0.5910	0.7088	0.4977	0.7027
3	0.5123	0.7165	0.4762	0.7027
4	0.4431	0.8008	0.3688	0.7027
5	0.3738	0.7816	0.2767	0.8784
6	0.3462	0.8659	0.2406	0.9459
7	0.3083	0.8697	0.2159	0.9054
8	0.2608	0.9080	0.2174	0.9189
9	0.2772	0.8812	0.1841	0.9595
10	0.2376	0.9080	0.1721	0.9595
11	0.2499	0.9042	0.2297	0.8784
12	0.1847	0.9349	0.4349	0.7297
13	0.2430	0.9195	0.2376	0.8514
14	0.1677	0.9157	0.1550	0.9595
15	0.1759	0.9234	0.1311	0.9730
16	0.1446	0.9464	0.1276	0.9730
17	0.1510	0.9464	0.1705	0.9054
18	0.1747	0.9310	0.4791	0.7432
19	0.2090	0.9119	0.2081	0.9054
20	0.1379	0.9464	0.1204	0.9865
21	0.1098	0.9693	0.1233	0.9730
22	0.1030	0.9808	0.2531	0.8514
23	0.0959	0.9847	0.1239	0.9595
24	0.0639	0.9923	0.1300	0.9459
25	0.0535	0.9962	0.1190	0.9595
26	0.0480	0.9962	0.1287	0.9459
27	0.0435	0.9962	0.1486	0.9459
28	0.0393	0.9962	0.1299	0.9459
29	0.0428	0.9962	0.1159	0.9595
30	0.0299	0.9962	0.1323	0.9459
31	0.0284	0.9962	0.1247	0.9459
32	0.0243	0.9962	0.1181	0.9595
33	0.0286	0.9962	0.1668	0.9459
34	0.0272	0.9962	0.1627	0.9459
35	0.0204	1.0000	0.1209	0.9730
36	0.0222	0.9962	0.1200	0.9595
37	0.0250	0.9962	0.2100	0.9324
38	0.0212	0.9962	0.1417	0.9459

39	0.0151	1.0000	0.1286	0.9595
40	0.0133	1.0000	0.1918	0.9459
41	0.0102	1.0000	0.1330	0.9595
42	0.0110	1.0000	0.1454	0.9459
43	0.0105	1.0000	0.1839	0.9459
44	0.0093	1.0000	0.1404	0.9595
45	0.0072	1.0000	0.1602	0.9459
46	0.0056	1.0000	0.1482	0.9459
47	0.0055	1.0000	0.1533	0.9459
48	0.0051	1.0000	0.1553	0.9459
49	0.0049	1.0000	0.1669	0.9459
50	0.0049	1.0000	0.1507	0.9459

Table 12. Model evaluation results

	Precision	Recall	F1-score	Support
High Yield	0.96	0.98	0.97	440
Low Yield	0.99	0.98	0.99	1060
Accuracy			0.98	1500
Macro avg	0.98	0.98	0.98	1500
Weighted avg	0.98	0.98	0.98	1500

Total number of images for low yield was 440, among which 432 were accurately classified and only 8 images were misclassified. On the other hand, total numbers of images in high yield were 1,060 among which 1043 images were accurately classified and only 17 were misclassified. If we overlook on the performance of this model, total number of images were 1500, among which accurate classifications made by model are 1,475 and false predictions were only 25.

4. Conclusions

In this study work an advanced version of machine learning algorithm, CNN is used for the determination of the soybean plant whether it yields high or low The prediction of the yield of the soybean crop over a large geographical area is important because of the increase in the world population as well as effects of climate change on crop production. It is a tool used in the early management of crops since it predicts field-scale yields and also points out on within-field yield differences. This makes for precise application of fertilizers, water, and pesticides so as to improve yield and return on investment and reduce wastage and pollution. There are various techniques that have been employed in the prediction of crop yield for instance crop growth models field stocks, remote sensing, and image analysis. These approaches are applied in feed management of the world and in formulation of grain policies besides precision farming. Here, CNN was used to predict the soybean plant yield from the input data of RGB crop field images. The data was collected from Kaggle and the number of images were only 370 which is not adequate for training of CNN. To augment the data TensorFlow image generator was used which produced 7520 images which are helpful for training the model. Some of the operations included in the image preprocessing were: resizing the images from 3456x4608 to 224×224 and rescaling the images to fit the new size. These steps were very important to improve the applied CNN model, with the fewer resources and less time for training. The model was trained three times using the changes in hyperparameters that are made to the model as seen in the subsequent sections. The final version of model delivers good performance with the testing accuracy of 92. 50%, cross validation accuracy of 94.59% and the training accuracy of 100%.

In the future, we intend to merge a diverse and large dataset including remote sensing data and climate data for the estimation of soybean crop yield. In addition, a user-friendly interface will also be developed for real-time monitoring as well as to keep the farmer updated regarding estimated yields. To this end, we also intend to cautiously utilize deep learning network topologies, and utilize adaptive learning rates, to train on data clusters rather than the complete dataset for improved performance.

References

1. Abu, M. A., Indra, N. H., Rahman, A. H. A., Sapiee, N. A., & Ahmad, I. (2019). A study on image classification based on deep learning and tensorflow. International Journal of Engineering Research and Technology, 12(4), 563–569.
2. Alabi, T. R., Abebe, A. T., Chigeza, G., & Fowobaje, K. R. (2022). Estimation of soybean grain yield from multispectral high-resolution UAV data with machine learning models in West Africa. Remote Sensing Applications: Society and Environment, 27(May), 100782. https://doi.org/10.1016/j.rsase.2022.100782
3. Amankulova, K., Farmonov, N., Abdelsamei, E., Szatmari, J., Khan, W., Zhran, M., Rustamov, J., Akhmedov, S., Sarimsakov, M., & Mucsi, L. (2024). A Novel Fusion Method for Soybean Yield Prediction Using Sentinel-2 and PlanetScope Imagery. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 1–18. https://doi.org/10.1109/JSTARS.2024.3402114
4. Archana, S., & Senthil Kumar, P. (2023). A Survey on Deep Learning Based Crop Yield Prediction. Nature Environment and Pollution Technology, 22(2), 579–592. https://doi.org/10.46488/NEPT.2023.v22i02.004
5. Barbedo, J. G. A. (2023). Deep Learning for Soybean Monitoring and Management. Seeds, 2(3), 340–356. https://doi.org/10.3390/seeds2030026
6. Barbosa dos Santos, V., Santos, A. M. F. dos, & Rolim, G. de S. (2021). Estimation and forecasting of soybean yield using artificial neural networks. Agronomy Journal, 113(4), 3193–3209. https://doi.org/10.1002/agj2.20729
7. Bhadra, S., Sagan, V., Skobalski, J., Grignola, F., Sarkar, S., & Vilbig, J. (2024). End-to-end 3D CNN for plot-scale soybean yield prediction using multitemporal UAV-based RGB images. Precision Agriculture, 25(2), 834–864. https://doi.org/10.1007/s11119-023-10096-8
8. Deshwal, P., Sharma, K., & Moudgil, M. S. (2023). Plant Leaf Disease Detection using Machine Learning. International Journal for Research in Applied Science and Engineering Technology, 11(5), 5928–5932. https://doi.org/10.22214/ijraset.2023.52895
9. Duc, N. T., Ramlal, A., Rajendran, A., Raju, D., Lal, S. K., Kumar, S., Sahoo, R. N., & Chinnusamy, V. (2023). Image-based phenotyping of seed architectural traits and prediction of seed weight using machine learning models in soybean. Frontiers in Plant Science, 14(September), 1–15. https://doi.org/10.3389/fpls.2023.1206357
10. Eugenio, F. C., Grohs, M., Venancio, L. P., Schuh, M., Bottega, E. L., Ruoso, R., Schons, C., Mallmann, C. L., Badin, T. L., & Fernandes, P. (2020). Estimation of soybean yield from machine learning techniques and multispectral RPAS imagery. Remote Sensing Applications: Society and Environment, 20(April). https://doi.org/10.1016/j.rsase.2020.100397
11. Habibi, L. N., Watanabe, T., Matsui, T., & Tanaka, T. S. T. (2021). Machine learning techniques to predict soybean plant density using UAV and satellite-based remote sensing. Remote Sensing, 13(13). https://doi.org/10.3390/rs13132548
12. Herrero-Huerta, M., Rodriguez-Gonzalvez, P., & Rainey, K. M. (2020). Yield prediction by machine learning from UAS-based mulit-sensor data fusion in soybean. Plant Methods, 16(1), 1–16. https://doi.org/10.1186/s13007-020-00620-6
13. Hollard, L., Durigon, A., & Steffenel, L. A. (2022). Machine Learning Forecast of Soybean Yields on South Brazil. https://doi.org/10.3233/aise220028
14. Isinkaye, F. (2022). A Smartphone-based Plant Disease Detection and Treatment Recommendation System using Machine Learning Techniques. Transactions on Machine Learning and Artificial Intelligence, 10(1), 1–8. https://doi.org/10.14738/tmlai.101.1131
15. Jubery, T. Z., Carley, C. N., Singh, A., Sarkar, S., Ganapathysubramanian, B., & Singh, A. K. (2021). Using machine learning to develop a fully automated Soybean Nodule Acquisition Pipeline (SNAP). Plant Phenomics, 2021. https://doi.org/10.34133/2021/9834746
16. Kale, N., Gunjal, S. N., Bhalerao, M., Khodke, H. E., Gore, S., & Dange, B. J. (2023). Crop Yield Estimation Using Deep Learning and Satellite Imagery, International Journal of Intelligent Systems and Applications in Engineering, IJISAE, 2023(10s), 464–471. https://orcid.org/0000-0003-1814-59131,
17. Ko, J., Shin, T., Kang, J., Baek, J., & Sang, W. G. (2024). Combining machine learning and remote sensing-integrated crop modeling for rice and soybean crop simulation. Frontiers in Plant Science, 15(February), 1–12. https://doi.org/10.3389/fpls.2024.1320969
18. L Hoffman, A., R Kemanian, A., & E Forest, C. (2020). The response of maize, sorghum, and soybean yield to growing-phase climate revealed with machine learning. Environmental Research Letters, 15(9). https://doi.org/10.1088/1748-9326/ab7b22

19. Li, X., Xu, X., Xiang, S., Chen, M., He, S., Wang, W., Xu, M., Liu, C., Yu, L., Liu, W., & Yang, W. (2023). Soybean leaf estimation based on RGB images and machine learning methods. Plant Methods, 19(1), 1–16. https://doi.org/10.1186/s13007-023-01023-z
20. Lu, J., Fu, H., Tang, X., Liu, Z., Huang, J., Zou, W., Chen, H., Sun, Y., Ning, X., & Li, J. (2024). GOA-optimized deep learning for soybean yield estimation using multi-source remote sensing data. Scientific Reports, 14(1), 1–19. https://doi.org/10.1038/s41598-024-57278-6
21. Machine learning methods for precision agriculture with UAV imagery : a review. (2022). https://doi.org/10.3934/era.2022218
22. Miranda, M. C. de C., Aono, A. H., & Pinheiro, J. B. (2023). A novel image-based approach for soybean seed phenotyping using machine learning techniques. Crop Science, 63(5), 2665–2684. https://doi.org/10.1002/csc2.21032
23. Moeinizade, S., Pham, H., Han, Y., Dobbels, A., & Hu, G. (2022). An applied deep learning approach for estimating soybean relative maturity from UAV imagery to aid plant breeding decisions. Machine Learning with Applications, 7(July 2021), 100233. https://doi.org/10.1016/j.mlwa.2021.100233
24. Mohite, J. D., Sawant, S. A., Pandit, A., Agrawal, R., & Pappula, S. (2023). Soybean Crop Yield Prediction By Integration of Remote Sensing and Weather Observations. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives, 48(M-1–2023), 197–202. https://doi.org/10.5194/isprs-archives-XLVIII-M-1-2023-197-2023
25. Nandhini, N., & Shankar, J. G. (2020). Prediction of Crop Growth Using Machine Learning Based on Seed Features. Ictact Journal on Soft Computing, 2232–2236. https://doi.org/10.21917/ijsc.2020.0319
26. Ranđelović, P., Đorđević, V., Miladinović, J., Prodanović, S., Ćeran, M., & Vollmann, J. (2023). High-throughput phenotyping for non-destructive estimation of soybean fresh biomass using a machine learning model and temporal UAV data. Plant Methods, 19(1), 1–13. https://doi.org/10.1186/s13007-023-01054-6
27. Richetti, J., Judge, J., Boote, K. J., Johann, J. A., Uribe-Opazo, M. A., Becker, W. R., Paludo, A., & Silva, L. C. de A. (2018). Using phenology-based enhanced vegetation index and machine learning for soybean yield estimation in Paraná State, Brazil. Journal of Applied Remote Sensing, 12(02), 1. https://doi.org/10.1117/1.jrs.12.026029
28. Santos, L. (2024). Machine Learning-Based Soybean Yield Prediction And Optimizing LIDAR-Mounted Uav Efficiency. LSU Master's Theses. https://repository.lsu.edu/gradschool_theses/5828
29. SARKAR, S. (2023). Quantifying Soybean Phenotypes Using Uav Imagery and Machine Learning , Deep Learning Methods. University of Missouri-Columbia, July.
30. Shammi, S. A., Huang, Y., Feng, G., Tewolde, H., Zhang, X., Jenkins, J., & Shankle, M. (2024). Application of UAV Multispectral Imaging to Monitor Soybean Growth with Yield Prediction through Machine Learning. Agronomy, 14(4), 672. https://doi.org/10.3390/agronomy14040672
31. CAAFRIT, Plant Leaves for Image Classification. (2021). Kaggle. https://www.kaggle.com/datasets/csafrit2/plant-leaves-for-image-classification
32. Sreeja, S. P., Asha, V., Saju, B., Chandrakantbhai, P. P., Prabhasan, P., & Prasad, A. (2022). Cotton Plant Disease Prediction using Deep Learning. Proceedings of the 2022 3rd International Conference on Communication, Computing and Industry 4.0, C2I4 2022, March. https://doi.org/10.1109/C2I456876.2022.10051527
33. Sundaramoorthi, D., & Dong, L. (2019). Machine-Learning-Based Simulation for Estimating Parameters in Portfolio Optimization: Empirical Application to Soybean Variety Selection. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.3412648
34. Terliksiz, A. S., & Altylar, D. T. (2019). Use of deep neural networks for crop yield prediction: A case study of soybean yield in lauderdale county, Alabama, USA. 2019 8th International Conference on Agro-Geoinformatics, Agro-Geoinformatics 2019, 9–12. https://doi.org/10.1109/Agro-Geoinformatics.2019.8820257
35. Tian, Y., Yang, C., Huang, W., Tang, J., Li, X., & Zhang, Q. (2021). Machine learning-based crop recognition from aerial remote sensing imagery. Frontiers of Earth Science, 15(1), 54–69. https://doi.org/10.1007/s11707-020-0861-x
36. Tong, Y. S., Lee, T. H., & Yen, K. S. (2022). Deep Learning for Image-Based Plant Growth Monitoring: A Review. International Journal of Engineering and Technology Innovation, 12(3), 225–246. https://doi.org/10.46604/ijeti.2022.8865
37. Venturini, V., Walker, E., Fonnegra, M. D. C., & Fagioli, G. (2022). Effect of the net radiation substitutes on maize and soybean evapotranspiration estimation using machine learning methods. AgriScientia, 39(2), 1–17. https://doi.org/10.31047/1668.298x.v39.n2.37104

38. Worrall, G., Rangarajan, A., & Judge, J. (2021). Domain-guided machine learning for remotely sensed in-season crop growth estimation. Remote Sensing, 13(22), 1–23. https://doi.org/10.3390/rs13224605
39. Yang, P., Zhao, Q., & Cai, X. (2020). Machine learning based estimation of land productivity in the contiguous US using biophysical predictors. Environmental Research Letters, 15(7). https://doi.org/10.1088/1748- 9326/ab865f
40. Yu, H., Weng, L., Wu, S., He, J., Yuan, Y., Wang, J., Xu, X., & Feng, X. (2024). Time-Series Field Phenotyping of Soybean Growth Analysis by Combining Multimodal Deep Learning and Dynamic Modeling. Plant Phenomics, 6, 1–12. https://doi.org/10.34133/plantphenomics.0158
41. Zhang, Y., Yang, Y., Zhang, Q., Duan, R., Liu, J., Qin, Y., & Wang, X. (2023). Toward Multi-Stage Phenotyping of Soybean with Multimodal UAV Sensor Data: A Comparison of Machine Learning Approaches for Leaf Area Index Estimation. Remote Sensing, 15(1). https://doi.org/10.3390/rs15010007