# Enhanced Approach for Crop Prediction Using Convolution Neural Networks

Ch Sridevi
P.G Scholar
sridevichittimenu@gmail.com
Bonam Venkata Chalamayya Engineering College(A)
Odalarevu.
B Satyanarayana Murthy
Professor of CSE
murthy2007.b@gmail.com
Bonam Venkata Chalamayya Engineering College(A)
Odalarevu.

**Abstract**: Traditional soil analysis methods can be time-consuming and may not provide real-time insights. This paper introduces a deep learning-based approach for soil type classification and crop recommendation, enhancing efficiency and accuracy. Utilizing Convolutional Neural Networks (CNNs), the system analyzes soil images to identify distinct characteristics and classify soil types. Based on the detected soil type, it further suggests suitable crops. Designed as a user-friendly application for both web and mobile platforms, the system allows users to upload soil images and receive instant classification results along with crop recommendations. The paper's success will be evaluated through classification accuracy, the relevance of crop suggestions, and user feedback on usability and effectiveness.

Keywords: Soil Classification, Machine Learning, Deep Learning, CNN,

## 1.INTRODUCTION:

The paper operates within the domain of precision agriculture, an innovative approach that leverages technology to optimize farming practices and maximize crop yields while minimizing resources and environmental impact. Precision agriculture integrates various technologies, including sensors, IoT devices, satellite imagery, and data analytics, to provide farmers with actionable insights for decision-making[1].

In traditional agriculture, farmers often employ uniform practices across their fields, treating them as homogenous entities. However, variations in soil type, moisture levels, and other environmental factors can significantly impact crop growth and yield. Precision agriculture seeks to address this by enabling farmers to tailor their actions based on specific conditions within their fields, leading to more efficient resource utilization and improved productivity.

The paper focuses on two key aspects of precision agriculture: soil type detection and crop prediction. Accurate identification of soil types allows farmers to understand the unique characteristics of their fields, enabling them to make informed decisions regarding irrigation, fertilization, and crop selection[2]. Crop prediction models utilize historical data, weather forecasts, and soil information to recommend the most suitable crops for cultivation, taking into account factors such as soil type compatibility and climate suitability.

By harnessing deep learning methods, the paper aims to enhance the accuracy and efficiency of soil type detection and crop prediction processes. Deep learning algorithms, such as convolutional neural networks (CNNs) as well as recurrent neural networks (RNNs), are well-suited for analyzing large volumes of complex data, making them ideal for addressing the challenges inherent in precision agriculture.

Overall, the paper seeks to empower farmers with advanced tools and insights that enable them to optimize their farming practices, increase crop yields, and contribute to the sustainability of agricultural production. Through the application of deep learning techniques, the paper endeavors to revolutionize the way farmers manage their fields, ushering in a new era of data-driven decision-making in agriculture.

### 1.1 Soil Type Detection

Soil type detection refers to the process of identifying and categorizing different types of soil based on their physical, chemical, and biological properties. It is a crucial aspect of precision agriculture, as soil type plays a significant role in determining crop suitability, irrigation requirements, nutrient management, and overall agricultural productivity.

Several methods can be employed for soil type detection, including traditional soil testing techniques, remote sensing technologies, and advanced data analytics approaches. In recent years, the advent of machine learning and deep learning algorithms has provided new opportunities to improve the accuracy and efficiency of soil type detection processes.

Deep learning techniques, such as convolutional neural networks (CNNs), have shown promise in analyzing spatial data, such as satellite imagery and soil sensor readings, to automatically classify different soil types[3]. By training CNN models on labeled datasets containing examples of various soil types, these algorithms can learn to extract relevant features from the input data and make predictions about the soil type present in a given area.

### 1.2 Crop prediction

Crop prediction, also known as yield prediction or crop forecasting, refers to the process of estimating the potential yield or production of crops in a given area for a specific growing season. It is a critical aspect of agricultural management, as accurate crop predictions enable farmers to make informed decisions regarding planting schedules, resource allocation, and marketing strategies.

Crop prediction involves the integration of various data sources and predictive modeling techniques to forecast crop yields based on factors such as weather conditions, soil properties, historical yield data, crop management practices, and pest and disease incidence. Machine learning and statistical modeling approaches are commonly used for crop prediction, with the goal of capturing the complex interactions between these factors and their impact on crop growth and development.

### 1.3 Deep Learning

Deep learning which is alternatively called deep structured learning is a division of machine learning based on ANN with representation learning. The word “deep” indicates the multiple numbers of layers in which data transformations occurs for reforming the data into high-level features. Deep learning systems specifically have a significant credit assignment path (CAP) depth. The series of transformations leading from input to output makes up the CAP. CAPs describe the relationships between input and output that might be causative[4]. The depth of the CAPs for a feed-forward neural network is equal to the network's depth and is equal to the number of hidden layers plus one (as the output layer is also parameterized). Unsupervised, semi-supervised, and supervised learning are all possible. Each degree of deep learning learns how to change the incoming data into a tad more abstract and composite representation.

A computer model learns to carry out categorization tasks directly from images, text, or sound using deep learning. Modern precision can be attained by deep learning models, sometimes even outperforming human ability. Using a sizable collection of labelled data, models are trained. There are numerous layers in neural network architectures. Numerous labelled data sets are necessary for deep learning. It calls for a lot of computational power. Deep learning is effectively supported by the parallel design of high-performance GPUs. When combined with clusters or cloud computing, this would enable development teams to reduce training time for a deep learning network from weeks to hoursorless.

### 1.4 Machine Learning

Algorithms for machine learning identify patterns in data and gain knowledge through generalization. ML models are trained on training and validation set and then their performance are checked on the test set. A machine learning workflow starts with relevant features being manually extracted from images. The features are then used to create a model that categorizes the objects in the image. Without using a preexisting equation as a model, machine learning algorithms employ computer techniques to "learn" information directly from data. As there are more samples available for learning, the algorithms adapt to their performance. Unsupervised learning, which identifies hidden patterns or intrinsic structures in input data, and supervised learning, which trains a model on known input and output data to predict future outputs, are the two types of approaches used in machine learning.

## 2. LITERATURE SURVEY

### 2.1 Crop Yield Prediction Using Effective Deep Learning and Dimensionality Reduction Approaches for Indian Regional Crops:

Crop yield prediction is a critical aspect of agricultural planning and decision-making, especially in a country like India where agriculture plays a significant role in the economy. Traditional methods of yield prediction often fall short in accuracy due to the complex interplay of various factors affecting crop growth. The study explores the application of deep learning techniques combined with dimensionality reduction to enhance the prediction accuracy for regional crops in India[5].

Climate Data: Temperature, rainfall, humidity, and other meteorological parameters. Soil Data: Soil type, pH, nutrient levels, and moisture content. Agronomic Practices: Sowing date, irrigation patterns, fertilizer usage, and crop rotation. Remote Sensing Data: Satellite imagery providing insights into vegetation indices, land use, and cover type.

The core algorithm employed in this study is a deep learning model, specifically a Convolutional Neural Network (CNN), tailored for the task of crop yield prediction. The CNN model is chosen for its ability to automatically capture spatial hierarchies in the input data, which is particularly beneficial when dealing with complex agricultural datasets. Additionally, dimensionality reduction techniques such as Principal Component Analysis (PCA) are applied to reduce the computational complexity and enhance the model's efficiency by focusing on the most significant features.

### 2.2 Soil Classification Using Deep Learning Techniques:

Soil classification is crucial for various applications, including agriculture, construction, and environmental management. Accurate soil classification helps in determining soil properties and suitability for specific uses. The study explores the application of deep learning techniques to improve the accuracy and efficiency of soil classification.

Soil Texture: Sand, silt, and clay composition[6].

Soil Color: Color indicators which can imply organic content and soil type.

Soil Chemical Properties: pH, nutrient levels (nitrogen, phosphorus,

### 2.3 Convolutional Neural Networks Based Classifications of Soil Images:

Soil classification is a fundamental task in agriculture, environmental science, and geotechnical engineering. Traditional soil classification methods, which rely on manual sampling and analysis, are time-consuming and prone to human error. The study explores the use of Convolutional Neural Networks (CNNs) to automate and improve the accuracy of soil image classification. By leveraging deep learning techniques, the research aims to provide a more efficient and precise approach to soil classification.
Soil Texture: Visual patterns indicating the proportions of sand, silt, and clay.
Soil Color: Variations in color that can signify different soil types and properties.
Soil Structure: Physical arrangement of soil particles visible in the images.
Image Metadata: Information such as location, lighting conditions, and camera settings.
The core algorithm employed in this study is the Convolutional Neural Network (CNN). CNNs are well-suited for image classification tasks due to their ability to automatically learn spatial hierarchies and patterns in the data.
Convolutional Layers: To extract features from the soil images by applying various filters. Pooling Layers: To reduce the dimensionality and computational complexity while preserving important features. Fully Connected Layers: To combine the features and perform the final classification. The model is trained on a large dataset of labeled soil images, with preprocessing steps such as normalization and data augmentation applied to improve robustness and accuracy[7].

**2.4 Machine Learning Based Soil-Type Classification:**

Soil-type classification is crucial for various applications, including agriculture, construction, and environmental management. Traditional methods of soil classification involve manual sampling and laboratory testing, which are time-consuming and labor-intensive. The study explores the use of machine learning techniques to automate and improve the accuracy of soil-type classification. This approach aims to provide a more efficient, scalable, and precise method for classifying soils, particularly in regions like Ethiopia.
Support Vector Machine (SVM): Effective for high-dimensional spaces and commonly used for classification tasks.
Random Forest (RF): An ensemble learning method that operates by constructing multiple decision trees and outputting the mode of the classes.
K-Nearest Neighbors (KNN): A simple, instance-based learning algorithm that classifies a sample based on the majority class among its k-nearest neighbors[8].
Neural Networks: Basic neural networks are used to capture complex patterns in the data.
The model training involves preprocessing steps such as normalization and feature selection to enhance model performance. Cross-validation techniques are used to ensure the robustness and generalizability of the models.

**2.5 Machine Learning Techniques in Crop Recommendation based on Soil and Crop Yield Prediction System:**

Crop recommendation and yield prediction are essential for optimizing agricultural productivity and ensuring food security. Traditional methods often rely on empirical knowledge and historical data, which may not be sufficient to account for the complex interactions between soil properties, climate conditions, and crop characteristics[10]. The reviews examines various machine learning techniques used to enhance crop recommendation systems and predict crop yields based on soil and environmental data. Decision Trees and Random Forests: Useful for handling non-linear relationships and feature importance analysis. Support Vector Machines (SVM): Effective for high-dimensional data and robust to overfitting. Artificial Neural Networks (ANN): Capable of capturing complex patterns and interactions in the data[9].
K-Nearest Neighbors (KNN): Simple and effective for classification tasks. Ensemble Methods: Combining multiple algorithms to improve accuracy and robustness. The review emphasizes the importance of feature selection, data preprocessing, and model evaluation

techniques, such as cross-validation and hyperparameter tuning, to enhance model performance[11].

## 3.IMPLEMENTATION

3.1 System Architecture:

3.1.1 Convolutional Neural Network (CNN):

Convolutional Neural Network is a part of Deep Learning. A Convolutional Neural Network consists of multiple layers of artificial neurons. In this, each group of neurons in a layer is interconnected with every other neuron in the layer below it. The output layer, which is the final layer, is a fully connected layer.

The predictions are represented in this output layer. In Image recognition, a Convolutional Neural Network (CNN) is a type of feed-forward artificial neural network in which the connectivity pattern between its neurons is inspired by the organization of animal visual cortex, whose individual neurons are arranged in such a way that responds to overlapping regions tiling the visual field. It uses a complex architecture composed of stacked layers in this is very well suited to categories the photographs. This architecture is reliable and sensitive to each feature contained in the images for multi-class classification.

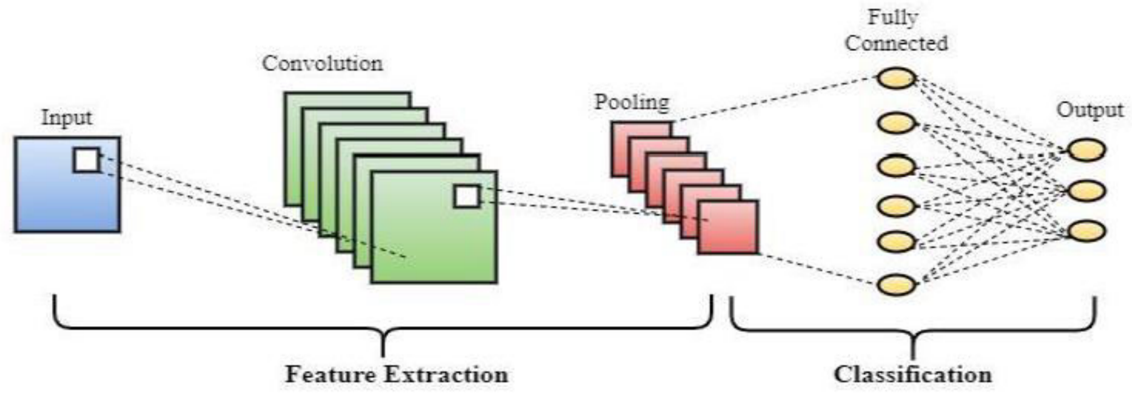


**Figure 1:** CNN Architecture

**A convolutional neural network typically consists of three layers, which are as follows:**

**Input**: The image will contain the raw pixel values ([32x32x3]) if it has 32 widths and 32 heights and three R, G, and B channels.

**Convolution**: In order for each neuron to produce a dot product between weights and the small input volume region they are actually linked to, it computes the output of those neurons that are associated with local input regions. For example, if we choose to incorporate 12filters, then it will result in a volume of [32x32x12].

**ReLU Layer**: It is specifically employed to apply an element-by-element activation function, such as a max (0, x) holding at zero. It yields ([32x32x12]), which pertains to the volume's size remaining constant.

**Pooling**: Using this layer, [16x16x12] volume is produced through a down sampling operation along the spatial dimensions (width, height). Locally Connected: It can be described as a typical layer of a neural network that computes class scores after receiving input from the layer above, producing a 1-dimensional array with the same size as the number of classes.

As the pooling layer's primary function is to ensure that our pictures have spatial invariance, we will next apply it to our convolutional layer to build a pooled feature map from each individual feature map. Additionally, it aids in reducing the size of our photos and preventing overfitting of our data. We will next flatten every image in our pool into a single long vector or column of all of these variables before feeding these values into our artificial neural network. The final output will be produced by feeding it into the locally linked layer

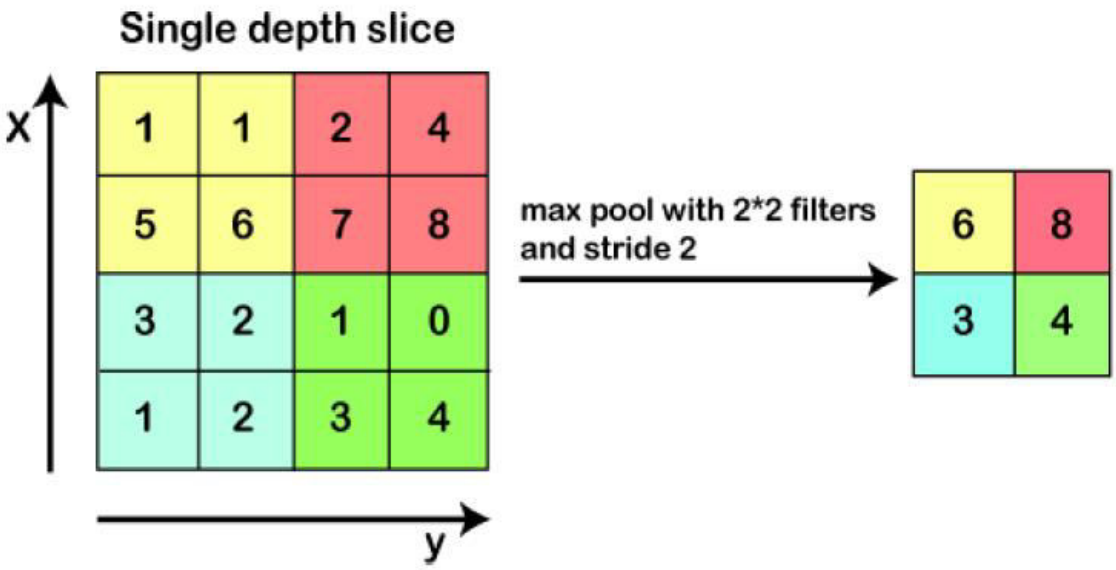


**Figure 2:** Pooling Operation

3.1.2 Logistic Regression

To frame a binary output, it is utilised (Varghese, 2018). Binary, multi-, and ordinal logistic regression are the three different kinds. In binary logistic regression, the outcome can be either pass or fail. Multiple outputs, such as cats, dogs, and sheep, are referred to as multi logistic regression. With ordinal logistic regression, there are numerous outputs arranged in descending order from low to medium to high.

3.1.3 Random Forest

The algorithm Random Forest is frequently employed in classification and regression issues. This method takes a long time yet produces accurate results. Using random forests, which are constructed from smaller data sets and deliver conclusions based on average or majority rating, prevents overfitting. It chooses observations at random, creates a decision tree, and takes the average outcome. It makes no use of any formulae It is a classifier that employs many decision trees on various subsets of the supplied dataset and averages the outcomes to improve the expected accuracy of the dataset.

3.1.4 Space and Time Complexity

Time Complexity:

The time complexity of a CNN is primarily determined by the number of operations performed during the forward and backward passes. The forward pass involves convolutions, pooling, and fully connected operations. The backward pass involves gradient computations for weight updates during training.

Convolutional Layers: The time complexity of a single convolutional layer with input size H×W (height and width of the input), C input channels, K kernels, kernel size F×F, and output channels O can be approximated as O(H×W×C×K×F×F).

Pooling Layers: The time complexity of pooling layers is generally considered relatively low compared to convolutions.

Fully Connected Layers: The time complexity of fully connected layers depends on the number of neurons in the layer. If the fully connected layer has N neurons, and the previous layer has M outputs, the complexity is approximately O (N×M).

Summing up the complexities across all layers in the network gives the overall time complexity for a single forward or backward pass. CNNs are often deep networks, so the overall complexity can be substantial, especially for larger input sizes and deeper architectures.

**Space Complexity:**

The space complexity of a CNN primarily refers to the memory required to store the network's parameters and intermediate activations during computation. It can vary significantly depending on factors such as the number of layers, the number of neurons per layer, and the input size.

**Parameters**: The space required to store the learnable parameters (weights and biases) of each layer in the network.

**Intermediate Activations**: During the forward pass, intermediate feature maps are computed at each layer. The space complexity depends on the size of these feature maps.

**Gradients**: During the backward pass (during training), gradients with respect to the parameters and activations need to be stored. The space complexity depends on the size of these gradient tensors.

**3.2 Algorithm:**

Convolutional Neural Network (CNN)

**i. Initialization:**

Import necessary libraries (TensorFlow/Keras, NumPy, OpenCV, etc.).
Load and preprocess the dataset.

**ii. Model Architecture:**

Define the CNN structure using sequential or functional API.
Add layers (Conv2D, MaxPooling2D, Flatten, Dense).
Specify activation functions (ReLU for hidden layers, Softmax for output layer).

**iii. Compilation:**

Choose an optimizer (e.g., Adam).
Define the loss function (categorical cross-entropy).
Set evaluation metrics (accuracy).

**iv. Training:**

Train the model on the training set using the fit() method.
Validate the model on the validation set to monitor overfitting.
Save the best model using checkpoints.

**v. Evaluation and Testing:**

Evaluate the model on the test set using the evaluate() method.
Fine-tune the model if necessary.

**vi. Prediction:**

Load the trained model.
Use the predict () method to make predictions on new soil images.
Post-process the results to display soil type and crop recommendations.

**3.3 Technique's:**

**i. Image Augmentation**

- Rotation: Rotating images by random angles.
- Flipping: Horizontal and vertical flipping.
- Scaling: Randomly zooming in and out.
- Translation: Shifting images horizontally or vertically.

**ii. Feature Extraction**

- Convolutional Layers: Using filters to detect edges, textures, and patterns.
- Pooling Layers: Reducing the spatial dimensions while preserving important features.

**iii. Model Optimization**
 Batch Normalization: Normalizing inputs of each layer to speed up training.
 Dropout: Randomly dropping units to prevent overfitting.
 Learning Rate Schedulers: Dynamically adjusting the learning rate during training.
**iv. Evaluation Metrics**
 Accuracy: Overall correctness of the model.
 Precision: Correctly predicted positive observations / Total predicted positives.
 Recall: Correctly predicted positive observations / all actual positives.
 F1-Score: Harmonic mean of precision and recall.
 Confusion Matrix: Matrix showing true positives, false positives, true negatives, and false negatives.
**v. Deployment**
 REST API: Creating endpoints for model predictions.
 Containerization: Using Docker to containerize the application for easy deployment.
 Cloud Deployment: Hosting the model on cloud platforms (AWS, Azure, GCP) for scalability and accessibility.

The Model Training Module is crucial for developing a robust CNN model capable of accurately classifying soil types and predicting suitable crops, ultimately aiding in agricultural decision-making.

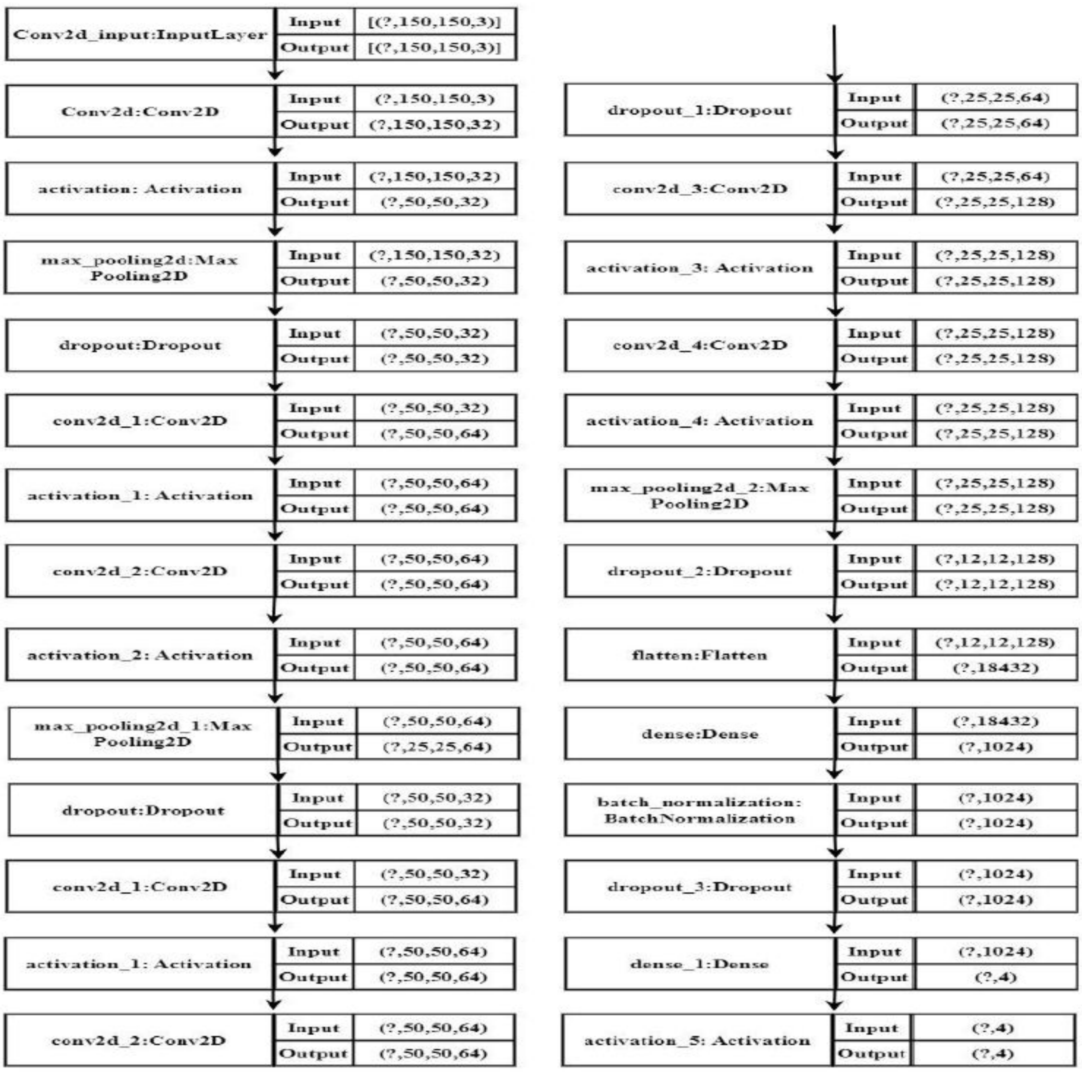


**Figure 4:** CNN Model Implementation

**STEP-1:** Input Layer (Conv2d_input)
Input: (?, 150, 150, 3)
Output: (?, 150, 150, 3)
Explanation: This is the input layer, where the input image has a dimension of 150x150 pixels with 3 channels (RGB). The question mark (?) represents the batch size, which can vary.
**STEP-2:** First Convolutional Layer (Conv2D)
Input: (?, 150, 150, 3)
Output: (?, 150, 150, 32)
Explanation: This layer applies 32 convolutional filters (kernels) of size 3x3 to the input image. It extracts local features such as edges, corners, and textures from the image while preserving the spatial dimensions.
**STEP-3:** Activation Layer (ReLU)
Input: (?, 150, 150, 32)
Output: (?, 150, 150, 32)
Explanation: The Rectified Linear Unit (ReLU) activation function is applied element-wise. It replaces all negative pixel values with zero, allowing the network to learn complex patterns.
**STEP-4:** First MaxPooling Layer
Input: (?, 150, 150, 32)
Output: (?, 50, 50, 32)
Explanation: MaxPooling reduces the spatial dimensions by selecting the maximum value from a 2x2 window, resulting in reduced computational complexity. Here, the output dimensions are halved, reducing it from 150x150 to 50x50 pixels.
**STEP-5:** Dropout Layer
Input: (?, 50, 50, 32)
Output: (?, 50, 50, 32)
Explanation: Dropout helps prevent overfitting by randomly dropping a fraction of neurons (typically during training), which forces the model to learn more robust features.
**STEP-6:** Second Convolutional Layer (Conv2D)
Input: (?, 50, 50, 32)
Output: (?, 50, 50, 64)
Explanation: A second convolutional layer with 64 filters of size 3x3 is applied to extract more complex features from the image.
**STEP-7:** Activation Layer (ReLU)
Input: (?, 50, 50, 64)
Output: (?, 50, 50, 64)
Explanation: ReLU activation is applied again, ensuring non-linearity and allowing the network to capture more intricate features.
**STEP-8:** Third Convolutional Layer (Conv2D)
Input: (?, 50, 50, 64)
Output: (?, 50, 50, 64)
Explanation: Another convolutional layer, which keeps the number of filters at 64, further refines the features learned by the previous layers.
**STEP-9**: Activation Layer (ReLU)
Input: (?, 50, 50, 64)
Output: (?, 50, 50, 64)
Explanation: A ReLU activation function is applied again to introduce non-linearity to the learned features.
**STEP-10:** Second MaxPooling Layer
Input: (?, 50, 50, 64)
Output: (?, 25, 25, 64)

Explanation: The second max-pooling layer reduces the spatial dimensions of the feature maps by taking the maximum value in each 2x2 region, reducing the dimensions from 50x50 to 25x25.
**STEP-11:** Second Dropout Layer
Input: (?, 25, 25, 64)
Output: (?, 25, 25, 64)
Explanation: Another dropout layer is applied to prevent overfitting by randomly dropping out units, improving generalization on unseen data.
**STEP-12:** Flatten Layer
Input: (?, 25, 25, 64)
Output: (?, 18432)
Explanation: The Flatten layer reshapes the 3D feature maps (25x25x64) into a 1D vector with 18,432 units. This prepares the data for the fully connected layers.
**STEP-13:** Fully Connected Layer (Dense)
Input: (?, 18432)
Output: (?, 1024)
Explanation: A fully connected (dense) layer with 1024 neurons is used to combine all the features learned by the previous layers. It allows the model to make predictions based on the feature vector.
**STEP-14:** Batch Normalization
Input: (?, 1024)
Output: (?, 1024)
Explanation: Batch normalization normalizes the output of the dense layer, ensuring that the activations have a stable distribution. This helps in speeding up training and improving model performance.
**STEP-15:** Third Dropout Layer
Input: (?, 1024)
Output: (?, 1024)
Explanation: A dropout layer is applied again to avoid overfitting in the fully connected layer.
**STEP-16:** Second Fully Connected Layer (Dense)
Input: (?, 1024)
Output: (?, 4)
Explanation: This is the final dense layer, which reduces the 1024 units to 4 output units, corresponding to the number of classes in your classification task (e.g., different soil types).
**STEP-17:** Activation Layer (Softmax)
Input: (?, 4)
Output: (?, 4)
Explanation: The softmax activation function is applied to output class probabilities, with each value in the output array representing the predicted probability for each class.
The CNN model processes the image through a series of convolutional, activation, pooling, and dropout layers to extract hierarchical features. Flattening is performed to convert the 3D feature maps into a 1D feature vector, followed by fully connected layers to make predictions.
Dropout and batch normalization are applied throughout the model to improve generalization and training stability, reducing the risk of overfitting.
This architecture is designed to efficiently process image data for classification tasks like soil type prediction, where features such as texture, pattern, and color need to be captured at different levels of granularity.

## 4.RESULTS

The soil classification training confusion matrix in the image presents an evaluation of the CNN model's performance in categorizing four distinct soil types: Red, Black, Alluvial, and

Peat. Each row corresponds to the true class, while each column represents the predicted class. Ideally, most of the values would lie along the diagonal, indicating correct predictions. In this matrix, Black soil shows the most accurate classification, with 172 samples correctly identified out of 197. However, Red soil suffers from significant misclassification, with only 1 correct prediction out of 191, and the rest mostly misclassified as Black (130) or Alluvial (60). The Alluvial soil shows moderate performance, with 130 correct classifications but a notable number of samples (65) misclassified as Black. Peat soil demonstrates the poorest performance, with only 2 out of 217 samples correctly identified, and a high number of samples being predicted as Black (129) or Alluvial (86). This high level of confusion between Peat, Black, and Alluvial soils suggests that the CNN model is struggling to distinguish between them, potentially due to similar texture or color features in the dataset. This highlights areas where the model could benefit from further refinement, such as improved feature extraction or balancing the dataset. The confusion matrix thus provides a clear visual summary of the model's strengths (e.g., accurate Black soil classification) and weaknesses (e.g., misclassifying Red and Peat soils).

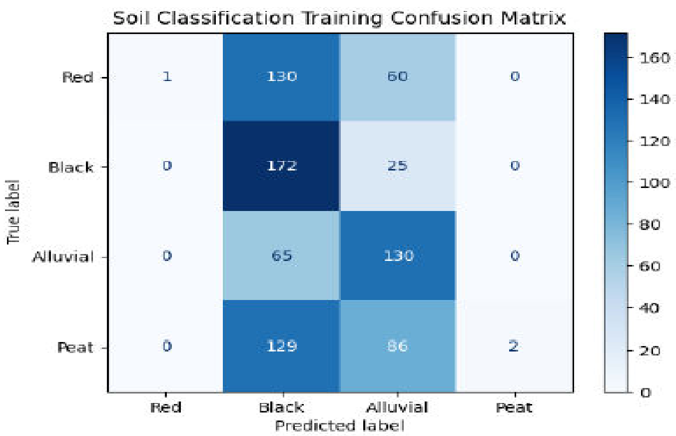


**Figure 5:** Soil Classification Training

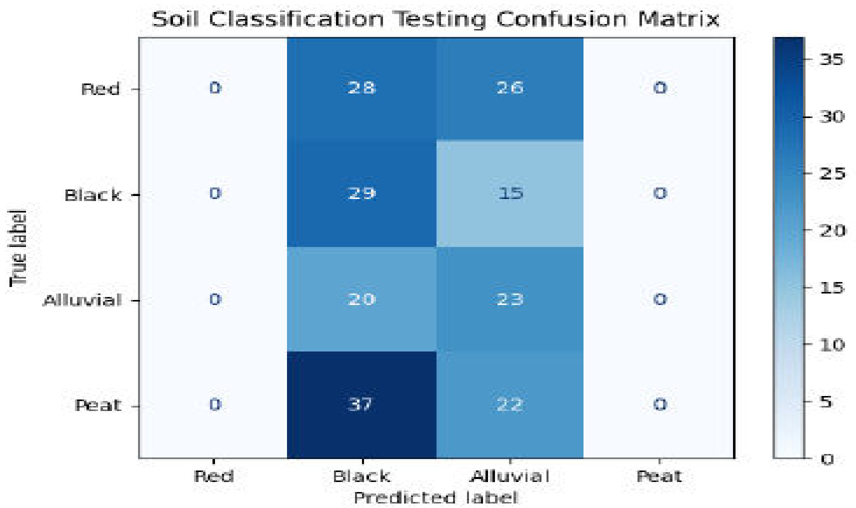


**Figure 6:** Soil Classification Testing

The image shows a confusion matrix for a soil classification model, which evaluates its erformance on testing data. The model predicts four classes: Red, Black, Alluvial, and Peat soils. The matrix compares the true labels (actual soil types) against the predicted labels.

**Training Process:**

Data Collection: Gather historical data on soil types, crops, and environmental conditions.
Feature Engineering: Extract and preprocess relevant features for model input.
Model Training: Train the model using supervised learning techniques on historical data.
Validation: Tune model parameters using validation data to optimize performance.

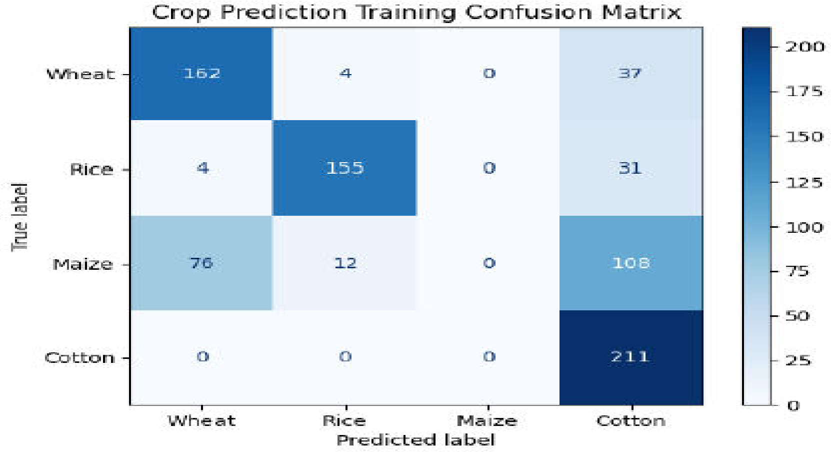


**Figure 7:** Crop prediction Training

The image offerings a confusion matrix for a crop classification model, which evaluates its performance across various crop types. The model predicts five classes: Wheat, Rice, Maize, Barley, and Sorghum, comparing true labels with predicted labels to assess accuracy.

**Design Considerations:**

Security: Ensure secure data exchange and API access.

Scalability: Design integration processes to handle large data volumes and multiple connections.

Flexibility: Allow for easy addition of new integration points as needed.

The Integration enables the soil type detection and crop prediction system to interact with a wide range of external systems and tools, enhancing its functionality and usability in the agricultural ecosystem.

**Accuracy and Loss in graphical representation:**

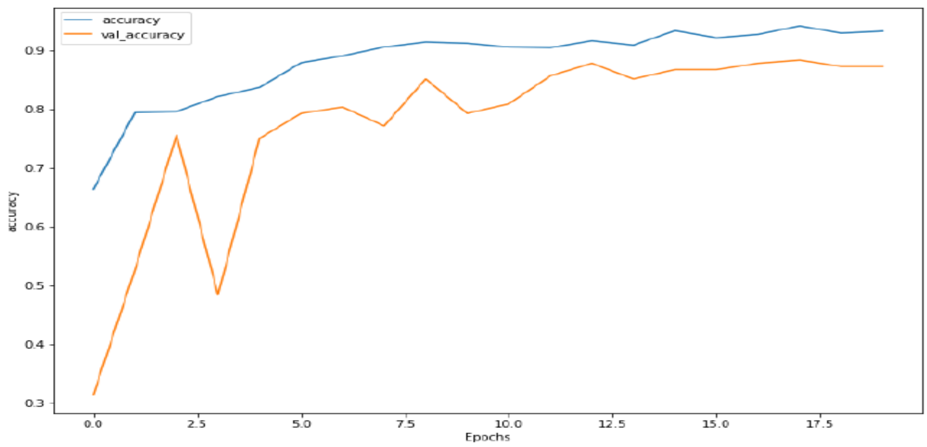


**Figure 9:** Training Accuracy Graph

## 5. CONCLUSION AND FUTURE SCOPE

We investigated the challenges of implementing drug tracking within pharmaceutical supply chains in an effort to address the substantial issue of counterfeit pharmaceuticals. We have developed and piloted a medical supply chain drug-tracking and tracing system that utilizes blockchain technology. Taken specifically, our proposed solution utilizes the cryptographic foundations of blockchain technology to accomplish immutable logs of supply chain events, and it utilizes smart contracts within the Ethereum block chain to accomplish automated recording of events that are accessible to all stakeholders. We proved that our proposed solution would help consumers save money on gas by reducing the number of procedures that the smart contract would initiate. Moreover, our proposed approach safeguards against hostile attempts to compromise the confidentiality, availability, and nonrepudiation of financial transactions— essential features in intricate, multi-party contexts such as the pharmaceutical supply chain— according to the results of the conducted security study.

While the current model achieves satisfactory results, there are several areas for future improvement and expansion. One potential enhancement is the incorporation of additional soil parameters, such as nutrient content and moisture levels, to provide a more comprehensive analysis of soil health. Integrating multispectral or hyperspectral image data could also improve the model's accuracy by capturing more detailed information about soil properties.

Future work could also explore the use of transfer learning by leveraging pre-trained models to improve classification accuracy, especially when dealing with limited data. Additionally, implementing a feedback loop where farmers can input their observations and results back into the system can help in continuously refining and updating the model. Developing a mobile application with offline capabilities would make this tool more accessible to farmers in remote areas with limited internet connectivity.

## REFERENCES

[1] Leelavathi Kandasamy Subramaniam, Rajasenathipathi Marimuthu "Crop yield prediction using effective deep learning and dimensionality reduction approaches for Indian regional crops" e-Prime - Advances in Electrical Engineering, Electronics and Energy 2024.

[2] D. Sivabalaselvamani, Rahunathan Loganathan, Nanthini Kempaiyan, T. Harshini "Soil Classification using Deep Learning Techniques" 2023 Second International Conference on Augmented Intelligence and Sustainable Systems (ICAISS).

[3] M.G. Lanjewar, O.L. Gurav "Convolutional Neural Networks based classifications of soil images" Multimedia Tools and Applications vol. 81, issue 7, Mar 2022.

[4] E. Enawugaw and Y. Yayeh, "Machine Learning Based Soil-Type Classification," 2023 International Conference on Information and Communication Technology for Development for Africa (ICT4DA), Bahir Dar, Ethiopia, 2023, pp. 165-170

[5] S. S. B, Anusha, A. Shetty, R. R. Shetty, B. A. D. Alva and A. D. Shetty, "Machine Learning Techniques in Crop Recommendation based on Soil and Crop Yield Prediction System – Review," 2022 International Conference on Artificial Intelligence and Data Engineering (AIDE), Karkala, India, 2022, pp. 230-235

[6] K. Lohitha Reddy and A. P. Siva Kumar, "Machine Learning Techniques for Weather based Crop Yield Prediction," 2023 Third International Conference on Artificial Intelligence and Smart Energy (ICAIS), Coimbatore, India, 2023, pp. 1263-1268

[7] Hamzah, Mohammad Diqi, Antomy David Ronaldo "Effective Soil Type Classification Using Convolutional Neural Network" Vol. 3, No.1, Agust us 2021.

[8] Ladan Samadi and Hanan Samadi, "Soil Classification Modelling Using Machine Learning Methods" Conference Paper · March 2022.

[9] A. Lakshmanarao et al.,"Crop Yield Prediction using Regression Models in Machine Learning," 2023 2nd International Conference on Applied Artificial Intelligence and Computing (ICAAIC), Salem, India, 2023.
[10] H. Samadi, J. Hassanpour, “Developing the empirical models for predicting the EPB operating parameters in strong Limestone,” JIRAEG Journal, 2021, (in press).
[11] K. Lohitha Reddy and A. P. Siva Kumar, "Machine Learning Techniques for Weather based Crop Yield Prediction," 2023 Third International Conference on Artificial Intelligence and Smart Energy (ICAIS), Coimbatore, India, 2023, pp. 1263-1268
[12] O. Varun Kumar, A. Ajith, S. Václav “Metaheuristic design of feedforward neural networks: A review of two decades of research,” Engineering Applications of Artificial Intelligence. vol. 60, 2017, pp. 97– 116.
[13] Y. Li, H. Zhang, Q. Shen, “Spectral-spatial classification of hyperspectral imagery with 3D convolutional neural network,” Remote Sensing, vol. 9, no. 1, p. 67, 2017.
[14] AG Boden Working Group, “Manual of soil mapping,”. E. Schweizerbart science publishers, Stuttgart, 2005, Germany.