International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 10 | Oct - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

CropAI-SOS: AI-Powered Crop Health Monitoring, Weather Alert, Geo-Mapping, and Emergency Support System for Farmers

Chirag Bhayal, Tirth H. Patel, Aditi Vaishnav, Ankit Sah, and Rohit
Department of Computer Science and Engineering Parul University, Vadodara, India
chiragbhayal7@gmail.com, ptirth0526@gmail.com, aditivaishnav106@gmail.com,
sahankita581@gmail.com, rohitatiiit@gmail.com

***Abstract*—Early detection of crop diseases and nutrient deficiencies is important for improving crop yield, reducing losses, and helping farmers take timely action. While AI can aid in accurate crop diagnostics, there is a lack of available, easy-to-use mobile tools that allow farmers to not only make an accurate diagnosis, but receive support in real time. The proposed system, called CropAI-SOS, is a smart agriculture support system which assists farmers in problem identification via mobile application and further environmental and emergency support. In this system, a farmer takes or uploads the image of the crop leaf onto his/her cell phone, and the image is fed into an AI-based prediction module in order to detect the presence of a disease or nutrient deficiency. The application shows the problem that has been detected, how certain the application is about the problem, and a basic treatment recommendation. To make the system more practical it also incorporates weather data and geo-location mapping to enable farmers to see not only weather conditions but the location of their crops and the crop context. A special SOS module is included to enable farmers to generate instant requests for support of their crops to the admin side, if the detected problem is serious. All scan records, SOS requests and user information is kept in a cloud-based database for monitoring and future reference. An admin dashboard is also included that provides an overview of users, scan reports, emergency cases and updates case status. The proposed system is simple, scalable and easy to use and showcases the potential of integrating AI, mobile computing, weather data and cloud services to provide a practical farmer support solution.**

***Index Terms*—Crop health monitoring, Plant disease detection, Smart agriculture, Weather alert system, Geo-mapping, Emergency farmer support, Machine learning, Artificial intelligence.**

I. Introduction

In many developing countries agriculture is an important part of the economy and for a significant proportion of people it is the way of life. Plant diseases, irregular weather, lack of timely information and insufficient access to expert guidance are some of the challenges farmers face. These challenges can have adverse impacts on productivity of the crops and cause economic losses. Hence, the importance of advanced technologies like Artificial Intelligence and data-driven systems to empower the farmers to take better decisions in agriculture is on the rise.

The recent advances in smart agriculture have brought about a range of technological solutions to support monitoring of crop health, identify plant diseases, and examine the surroundings. Machine learning and artificial intelligence techniques are being widely used to process agricultural data and provide useful insights to farmers [3], [4], [11]. They can help to identify crop diseases early, predict weather conditions and recommend appropriate farming practices. However, many existing systems focus on a single aspect of agriculture such as disease detection or weather monitoring and do not provide a comprehensive support system for farmers.

In this research work we proposed CropAI-SOS: AI-Driven Crop Health Monitoring, Weather Alert, Geo Mapping and Emergency Support System for Farmers. The proposed system is designed to help the farmers by monitoring the crop conditions using artificial intelligence, identifying the plant diseases using image analysis and providing early weather alerts. Geo-mapping technology also pinpoints farm locations, and an emergency support feature allows farmers to call for help when they are in a pinch. The main goal of this work is to create a smart agricultural system to help farmers improve crop productivity, minimize losses, and make smarter decisions.

The CropAI-SOS platform is designed to provide a trusted and farmer-friendly solution that harnesses AI, machine learning and location-based technologies to solve the challenges of modern agriculture.

II. Literature Review

Modern technologies like Artificial Intelligence, Machine Learning and Data Analytics have been slowly helping agriculture. Many researchers have worked on systems that help farmers to monitor crop status, diagnose plant diseases and make better decisions for agricultural activities [1], [6], [7], [13]. Most of these systems are mainly focused on the detection of diseases on plant leaves using image processing and machine learning algorithms. The systems are able to determine if a particular disease is affecting the plant, by analyzing images of crops. These allow farmers to identify problems in crops at an early stage and minimize their losses [5], [15].

Some papers about the use of Machine Learning have been studied. Random Forest and Support Vector Machines (SVM) have been extensively applied in the analysis of agricultural data and increasing predictive accuracy [4]. These models can learn patterns from many image datasets and provide automated disease detection. Many of these systems, however, can only detect diseases, and fail to offer other services like weather warnings or location-based services for farmers.

Smart agriculture platforms combining several technologies have been investigated over the last years. In some systems, weather forecasting and crop monitoring are integrated for more appropriate recommendations for crop activities [8], [9], [19]. During extreme weather events like heavy rainfalls, droughts or temperature changes, weather alert systems enable farmers to take preventive action. These solutions improve planning and mitigating risks from weather variability and change.

Another important development in agricultural technology is the use of geo-mapping and location-based services. Geo-mapping is used in agriculture to locate farms, track crop density and help farmers manage their crops. Conditions based on region, and access region-based agricultural information [10], [12]. By integrating geographical data can be linked to crop monitoring systems, enabling farmers to get better familiar with surroundings that impact on their crops.

Nevertheless, though there have been great strides, existing agricultural systems often concentrate on a single aspect, either crop disease detection, or weather monitoring. No integrated platforms exist that integrate crop health monitoring, weather alerts, geo-mapping and emergency support into one system [14], [16]–[18], [20]. Therefore, the proposed CropAI-SOS system aims to address these limitations by integrating multiple agricultural support services into one platform to assist farmers in managing their crops more efficiently.

III. Background: Artificial Neural Networks and Convolutional Neural Networks

Before describing the proposed system, this section briefly reviews the two foundational Deep Learning models relevant to CropAI-SOS: the Artificial Neural Network (ANN) and the Convolutional Neural Network (CNN). This background clarifies why a CNN-based architecture, rather than a generic ANN, was selected for the crop disease detection module.

A. Artificial Neural Network (ANN)

An Artificial Neural Network (ANN) is a Deep Learning model inspired by the structure and functioning of the human brain. It consists of interconnected artificial neurons organized into an input layer, one or more hidden layers, and an output layer. The input layer receives the features to be put into the system, the hidden layer performs a weighted calculation and adds non-linearity through an activation function, and the output layer provides the output. This process aims to minimize prediction error.

ANN can learn relations which are non-linear and complicated, and they are used in many cases such as stock price prediction, weather forecasting, fraud detection, and diagnosis in the medical field. The ANN, however, on an image treats an image as a list of numbers, and ignores the spatial location of different pixels of the image. This constraint reduces the efficiency and accuracy of plain ANN architectures for classification problems like crop disease detection based on images.

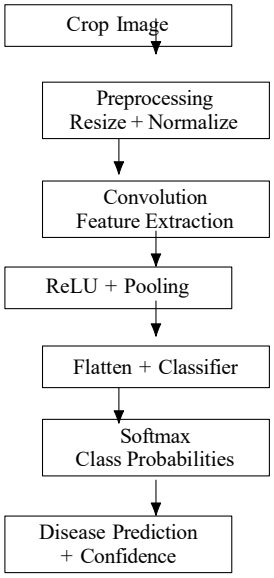


Fig. 1. CNN-based crop disease detection pipeline used in CropAI-SOS.

B. Convolutional Neural Network (CNN)

A Convolutional Neural Network (CNN) is a particular form of neural network, optimized for tasks that involve image processing and computer vision. A CNN unlike a traditional ANN, keeps the spatial structure of an image and learns hierarchical visual features such as edges, textures, colors, shapes and disease-specific features or patterns.

The typical CNN pipeline includes an input image layer, convolution layers for applying learnable filters to identify patterns in the local image, ReLU activation layers to add non-linearity, pooling layers to downsample spatial dimensions while keeping important information, a flatten layer to flatten the output feature maps into a one-dimensional vector, and fully connected layers followed by a Softmax layer for providing class probabilities. CNNs have been able to outperform plain ANNs on image classification tasks consistently [21], [22] because they are able to learn features directly from raw pixel data and preserve the spatial context. Hence, CropAI-SOS is designed using a CNN-based architecture for disease detection. Fig. 1 illustrates the image-processing sequence, from image acquisition and normalization to convolution, activation, classification, and the final disease prediction with its confidence score.

In the convolution stage shown in Fig. 1, local pixel values are combined with learnable filters to produce feature maps.

TABLE I
COMPARISON OF ANN AND CNN

Feature	ANN	CNN
Input	Numerical / flattened data	Images
Feature Extraction	Manual	Automatic
Spatial Information	Not preserved	Preserved
Best Suited For	Prediction, regression, tabular classification	Image classification, computer vision
Accuracy on Images	Moderate	Very High
Typical Use in This Project	Not used	Used (via MobileNetV2)

The following convolution expression describes the weighted sum and bias at each output location:

$$z_{i,j,k} = b_k + \sum_m \sum_n w_{m,n,k} x_{i+m,j+n}. \quad (1)$$

Here, x is the input feature map, w is a learnable filter, b_k is the bias of filter k, and $z_{i,j,k}$ is the resulting feature-map value. The following ReLU expression describes the activation stage in Fig. 1: negative feature responses are set to zero while positive responses are retained.

$$\mathrm{ReLU}(z) = \max(0, z). \quad (2)$$

Since the main requirement of CropAI-SOS is to classify leaf images into healthy and diseased, the CNN-based method is employed here.

IV. SYSTEM ARCHITECTURE

A. Architectural Overview

The design of the proposed CropAI-SOS system is modular. The architecture consists of (i) agricultural data collectors to gather agricultural information, (ii) intelligent models to process the agricultural information, and (iii) useful information generator and alert generation block to give useful information and alert to the farmers. The architecture of each module consists of a specific task which yields a result and which will be used as input to the next module in the processing sequence. The system starts by gathering information from different sources. Images of crops, weather, and geographical location data are processed and analyzed using machine learning models and supporting technologies. Based on the analysis, the system produces outputs such as crop health reports, disease detection results, warnings for weather conditions, and emergency notifications. The modular construction allows for the system to be easily expanded or upgraded as needed.

Fig. 2 shows how the farmer mobile application, FastAPI backend, CNN-based disease detection, weather and mapping services, Firebase storage, SOS support, and admin dashboard are connected.

The architecture also allows for a feedback mechanism to be introduced, such as farmers receiving alerts and recommendations so that they can take timely measures to protect their crops. By integrating crop monitoring, weather forecasting, geo-mapping, and emergency functions into one platform, the system forms a comprehensive agricultural support system.

B. Processing Pipeline

1) **Data Collection Layer:** The raw agricultural data is gathered from various sources in the system. This involves farming photos taken by farmers, environmental data including temperature and humidity, weather information from weather forecasting services, and geographical data of farms. These inputs are vital to know the crop condition and environmental elements that influence agricultural productivity.
2) **Data Preprocessing Layer:** After data collection, it is preprocessed to eliminate noise and prepare it for analysis. Crop images are resized and enhanced to improve image quality, and environmental and weather data are cleaned and structured for further processing. This ensures that the data fed to the machine learning models is uniform and appropriate for precise analysis. The normalization step represented in Fig. 1 scales image pixel values to a common range using

$$x_{\mathrm{norm}} = \frac{x - x_{\min}}{x_{\max} - x_{\min}}. \quad (3)$$

1) **Feature Extraction Layer:** Image processing techniques are used to detect patterns, textures, and colour variations on plant leaves. These extracted features are used for recognizing plant diseases or unhealthy plant conditions.
2) **Crop Disease Detection Layer:** This layer applies machine learning or deep learning models to the processed crop images. The extracted features are compared with known disease patterns in the trained model, which predicts whether the plant is healthy or has a specific disease. The classification stage in Fig. 1 converts the final CNN logits into class probabilities using

$$P(y = c \mid x) = \frac{e^{z_c}}{\sum_{j=1}^{C} e^{z_j}}. \quad (4)$$

Here, C is the number of output classes and z_c is the logit corresponding to class c. The following prediction expression selects the class with the highest Softmax probability:

$$\hat{y} = c \; P(y = c \mid x). \quad (5)$$

The early warning allows farmers to take preventive action against crop loss.

3) **Weather Monitoring and Alert Feature:** Data on rainfall, temperature, humidity and wind speed is analyzed to predict adverse weather conditions ahead of time. Farmers are notified about potential weather conditions that may affect their farming operations, so they can plan irrigation, harvesting or crop protection.
4) **Geo-Mapping and Location Analysis Layer:** This layer finds the geographical location of farms and visualizes them on a digital map using geo-mapping technology. This enables location-based agricultural services and region-specific recommendations.

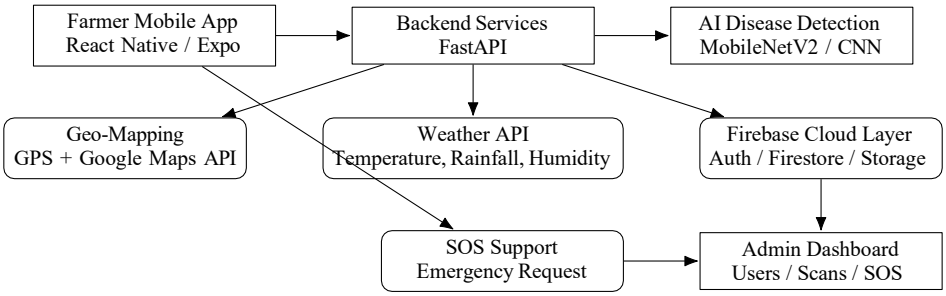


Fig. 2. Overall system architecture of the proposed CropAI-SOS platform.

5) **Emergency Support Module:** The emergency support module enables farmers to alert support services in case of agricultural emergencies, such as extreme crop damage or pests.
6) **Decision Support and Recommendation Layer:** This layer gives recommendations to farmers based on collected data and analysis results. These tips can contain disease treatment, management strategies, irrigation planning, and preventative measures depending on the weather.
7) **Reporting and Visualization Layer:** The final stage produces structured reports summarizing crop health status, identified disease, weather alerts and recommended solutions. The information is clearly presented to farmers and agricultural experts so that they can interpret the results and take necessary action.

V. Algorithm and Methodology

The CropAI-SOS is a multi-phase computational system that takes agricultural data and provides farmers with valuable information. The system captures crop images, weather data, and location details, followed by processing them through machine learning algorithms to identify diseases, generate alerts, and recommend solutions.

A. Phase 1: Data Collection and Preprocessing

This stage gathers crop images, weather information and farm location. The collected data is cleaned and prepared prior to feeding it to the machine learning model.

Algorithm 1 Data Collection and Preprocessing

Require: Crop images dataset, weather data, farm location data
Ensure: Cleaned and structured dataset
Initialize empty dataset *ProcessedData*
for all crop images *I* in dataset **do**
  Resize image to standard resolution
  Remove noise and normalize image values
  Store processed image in *ProcessedData*
end for
Collect weather parameters (temperature, humidity, rainfall)
Collect farm geographic coordinates
Combine image, weather, and location data
return *ProcessedData*

Algorithm 2 Image Feature Extraction

Require: Preprocessed crop images
Ensure: Feature vectors
for all image *I* in processed data **do**
  Convert image to numerical matrix
  Extract color features
  Extract texture features
  Extract shape and edge features
end for
Store extracted features into feature vector
return *FeatureVector*

B. Phase 2: Feature Extraction

Important visual features are extracted from crop images to identify disease symptoms such as spots, discoloration, or abnormal textures.

C. Phase 3: Crop Disease Detection Model

This phase uses a machine learning or deep learning model to classify crop images into healthy or diseased categories. During model training, the following categorical cross-entropy loss expresses the classification error by comparing the predicted class probabilities with the true labels:

$$L = -\sum_{c=1}^{C} y_c \log(\hat{p}_c). \tag{6}$$

where y_c is the true class indicator and $\hat{p}_c$ is the predicted probability for class c. Minimizing this loss encourages the model to assign higher probability to the correct disease class.

D. Phase 4: Weather Monitoring and Alert System

This phase analyzes weather data to generate alerts for farmers. The following threshold function formalizes the alert decision: an alert is activated when a measured parameter exceeds its configured threshold.

$$A(x) = \begin{cases} 1, & x > T, \\ 0, & x \le T. \end{cases} \tag{7}$$

Here, x is the measured weather parameter and T is its alert threshold. The following expressions specify the temperature,

Algorithm 3 Disease Classification Model

Require: Feature vectors or processed images
Ensure: Disease prediction results

```
Split dataset into Training (80%) and Testing (20%)
Initialize Convolutional Neural Network (CNN)
Train model using training dataset
for all test images X_test do
    Predict crop health condition
    if predicted label = healthy then
        Mark crop as Healthy
    else
        Mark crop as Diseased
    end if
end for
Calculate accuracy and performance metrics
return prediction results
```

Algorithm 4 Weather Alert Detection

Require: Weather data (temperature, humidity, rainfall)
Ensure: Weather alerts

```
Collect weather parameters from weather API
for all parameters do
    if temperature > threshold then
        Generate High Temperature Alert
    end if
    if rainfall > threshold then
        Generate Heavy Rain Alert
    end if
    if humidity > threshold then
        Generate High Humidity Alert
    end if
end for
Store alerts in notification system
Send alerts to farmers
```

rainfall, and humidity thresholds reported in the results:

$$T_{temperature} = 40^{\circ}C, \quad T_{rainfall} = 100\,mm, \quad T_{humidity} = 80\% \tag{8}$$

E. Phase 5: Geo-Mapping and Location Analysis

This phase maps farm locations and provides location-based agricultural insights. If a distance between two GPS coordinates is required, the following Haversine expression provides a mathematical method for estimating great-circle distance:

$$d = 2R \arcsin\sqrt{\sin^2\left(\frac{\Delta\phi}{2}\right) + \cos(\phi_1)\cos(\phi_2)\sin^2\left(\frac{\Delta\lambda}{2}\right)} \tag{9}$$

where R is the Earth's radius, ϕ_1, ϕ_2 are the latitudes, and $\Delta\lambda$ is the difference in longitude. This equation is included as the mathematical basis for coordinate-based distance analysis; the paper's reported geo-mapping result concerns farm locations displayed on the digital map.

Algorithm 5 Farm Location Mapping

Require: Farm GPS coordinates
Ensure: Digital farm map

```
Collect latitude and longitude coordinates
Connect to mapping service (Google Maps API)
Plot farm location on digital map
Display nearby agricultural information
return map visualization
```

Algorithm 6 Farmer Notification System

Require: Disease detection results, weather alerts
Ensure: Farmer notifications

```
for all crop analysis results do
    if disease detected then
        Send Disease Alert
        Suggest recommended treatment
    end if
end for
for all weather alerts do
    Send warning notification to farmer
end for
Generate crop health report
Deliver report through dashboard or mobile notification
```

F. Phase 6: Notification and Recommendation System

This phase generates alerts and recommendations based on crop health and weather conditions. Fig. 3 summarizes the overall methodology: collected images, weather readings, and GPS coordinates pass through preprocessing and disease detection before the relevant weather alert, emergency-support, mapping, and reporting paths are used. The diagram represents the proposed workflow rather than an additional experimental result.

G. Backend API Specification

The backend exposes a REST API through FastAPI. The primary endpoint is POST /predict, which takes a crop leaf image input as multipart/form-data, performs preprocessing and inference with the trained MobileNetV2 model, and returns a structured JSON response with predicted disease, confidence score, and treatment recommendation. A secondary GET /health endpoint reports backend availability, and an SOS endpoint accepts emergency support requests from the mobile application. The model is loaded once at server startup so repeated prediction requests do not incur reload overhead.

VI. Comparative Study of CNN Architectures

A range of CNN architectures were considered before finalizing the disease detection model. Table III summarizes the major CNN models relevant to image classification, their year of introduction, core technique, and typical use case, in order to justify the final architectural choice [23]–[26].

Among them, VGG16 [23], ResNet50 [24], DenseNet121 [25] and EfficientNet [26] provide good accuracy at the cost of

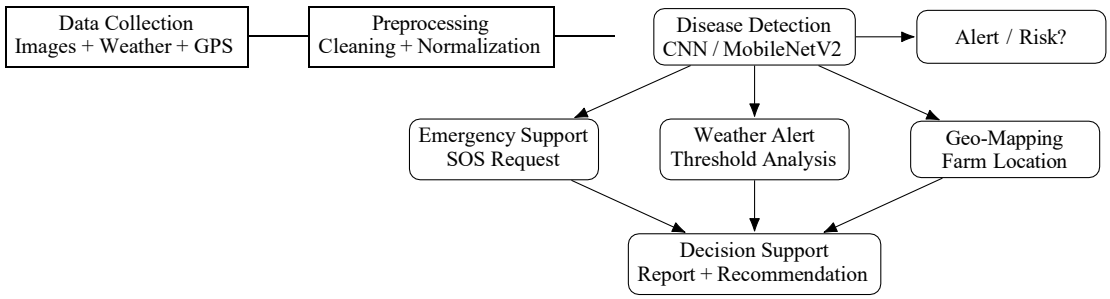


Fig. 3. End-to-end methodology and decision-support workflow of CropAI-SOS.

TABLE II
SAMPLE /PREDICT API RESPONSE

Field	Value
success	true
disease	Apple Apple scab
display name	Apple Scab
confidence	98.42
recommendations	Apply fungicide; Remove infected leaves

larger size and higher computation, which is less appropriate for direct deployment on a mobile client. MobileNetV1 first proposed the depthwise-separable-convolution method for efficient mobile inference. MobileNetV2 further extends this by adding inverted residual blocks and linear bottlenecks to increase the accuracy while keeping the model lightweight.

A. Justification for Selecting MobileNetV2

CropAI-SOS is a mobile-first application, the disease detection model should be efficient on resource-limited devices and provide high classification accuracy. Table IV compares MobileNetV2 to two commonly considered alternatives, ResNet50 and VGG16, on criteria most relevant to this deployment context.

MobileNetV2 was selected due to the following reasons: (1) it is designed for mobile and embedded applications, hence satisfying the low-latency, low-memory requirements of the React Native application; (2) it allows for Transfer Learning from ImageNet-pretrained weights, which reduces training time and improves accuracy on a modestly sized dataset like PlantVillage; (3) it is approximately 14 MB in size, much smaller than VGG16 or ResNet50, making on-device or backend deployment easier; (4) it provides fast inference, keeping the prediction latency low for the farmer; and (5) empirically, MobileNetV2 with transfer learning provides good accuracy in the 95%+ range reported in similar PlantVillage-based studies while still being practical to deploy. Heavier architectures such as ResNet50 or EfficientNet may give a slightly better accuracy but were considered impractical for this mobile deployment scenario, because of their larger size and slower inference.

B. Model Selection Guidance by Dataset Size

The scale of the dataset is also a factor when choosing a model for image classification. Table V contains popular architectures recommended for small, medium and large datasets, and the PlantVillage dataset used in this project (approximately 54,000 images for 38 classes) is placed within that spectrum.

The PlantVillage dataset is on the edge of the small-to-medium range. In this range, MobileNetV2 is still a good choice as the binding constraint of the project is mobile deployability and not raw dataset scale. Architectures suggested for medium or large datasets (e.g., ResNet50, EfficientNet-B3, Vision Transformers) would only improve accuracy slightly on a dataset of this size while significantly increasing model size and inference latency, which is in direct conflict with the mobile-first design goal of CropAI-SOS.

VII. RESULTS

This section presents the experimental results derived from the CropAI-SOS agricultural monitoring system. The evaluation includes the crop disease detection performance, the effectiveness of weather alerts, the accuracy of geo-mapping, and the generated recommendation outputs.

A. Crop Disease Detection Results

The disease detection module was tested on a dataset of images of plant leaves. The deep learning model, once trained, was able to classify images as either healthy or diseased. The evaluation metrics are defined from the confusion-matrix counts as follows. The accuracy expression below measures the overall proportion of correct predictions:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}. \quad (10)$$

The following precision expression measures the proportion of positive predictions that are correct:

$$Precision = \frac{TP}{TP + FP}. \quad (11)$$

The following recall expression measures the proportion of actual positive cases that are correctly detected:

$$Recall = \frac{TP}{TP + FN}. \quad (12)$$

TABLE III
COMPARISON OF MAJOR CNN ARCHITECTURES

Model	Year	Core Technique	Typical Use Case
LeNet-5	1998	Convolution + Pooling + Fully Connected	Digit recognition
AlexNet	2012	ReLU + Dropout + GPU training	Image classification
VGG16	2014	Stacked 3×3 convolution filters	Medical & agriculture imaging
GoogLeNet (Inception)	2014	Parallel multi-size filters (Inception module)	Image recognition
ResNet50	2015	Residual (skip) connections	Large-scale classification
DenseNet121	2016	Dense inter-layer connections	Medical & agriculture imaging
Xception	2017	Depthwise separable convolution	Image classification
MobileNetV1	2017	Depthwise separable convolution	Mobile devices
MobileNetV2 (used)	2018	Inverted residuals + linear bottlenecks	Mobile-based image classification

TABLE IV
MODEL SIZE AND SPEED COMPARISON

Model	Approx. Size	Speed	Mobile-Friendly
VGG16	~528 MB	Slow	No
ResNet50	~98 MB	Medium	No
MobileNetV2 (used)	~14 MB	Very Fast	Yes

TABLE V
MODEL SUITABILITY BY DATASET SIZE

Dataset Scale	Approx. Size	Recommended Models
Small	1,000–50,000	MobileNetV2, EfficientNet-B0, DenseNet121, VGG16
Medium (this project)	50,000–500,000	ResNet50, EfficientNet (B0–B3), DenseNet121, InceptionV3, Xception
Large	500,000+	ResNet101/152, EfficientNet-B4–B7, Vision Transformer (ViT), ConvNeXt, Swin Transformer

TABLE VI
MODEL PERFORMANCE EVALUATION

Model	Type	Acc.	Prec.	Rec.
SVM	Classical ML	0.78	0.74	0.81
CNN	Deep Learning	0.90	0.88	0.92

TABLE VII
WEATHER MONITORING ALERTS

Weather Condition	Threshold	Alert Generated
High Temperature	> 40°C	Heat Alert
Heavy Rainfall	> 100 mm	Flood Risk Alert
High Humidity	> 80%	Disease Risk Alert

The following F1-score expression combines precision and recall through their harmonic mean:

$$F1 = 2\frac{Precision \times Recall}{Precision + Recall} \quad (13)$$

Here, TP, TN, FP, and FN denote true positives, true negatives, false positives, and false negatives, respectively. Using the definitions above, the reported evaluation values are shown in Table VI. The results showed that the CNN-based deep learning model had higher precision and recall, and performed better in identifying crop diseases compared to conventional machine learning algorithms. The reported train/inference times were 0.45/0.009 s for SVM and 2.31/0.015 s for CNN.

B. Weather Alert System Results

The weather monitoring module was able to identify abnormal environmental conditions and generate alerts. Weather alerts assist farmers take preventive actions to protect crops against environmental threats.

C. Geo-Mapping and Location Monitoring

This feature facilitates location-based crop monitoring and region-specific recommendations. The geo-mapping module could map the farm locations and provide location-specific agricultural information.

D. Final Crop Monitoring Output

The system produced structured reports containing information on the health of crops and suggested actions. The crop diseases identified were properly classified and mapped to suitable treatment recommendations thus enabling farmers to take timely corrective measures.

VIII. OBSERVATIONS / ANALYSIS

This section presents the analysis and key observations from the experimental evaluation of CropAI-SOS. The observations highlight the performance of the different modules of the system and their contribution to the improvement of agricultural monitoring and decision-making.

A. Observation 1 — AI-Based Disease Detection Improves Accuracy

The plant disease detection model based on deep learning performs better than traditional image analysis techniques. The trained model was more accurate in classifying plant leaves into healthy and diseased classes. Deep learning models, especially Convolutional Neural Networks (CNN), are well suited to learn visual patterns like leaf texture, color variations and disease spots.

The results showed that models driven by AI performed better in the recognition of crop diseases in real settings

TABLE VIII
GEO-MAPPING OUTPUT SUMMARY

Output Type	Result
Total Farms Mapped	50
Location Accuracy	High
Map Visualization	Interactive Map Generated

TABLE IX
EXAMPLE CROP INCIDENT DETECTION

Incident Type	Evidence	Severity	Output Generated
Leaf Spot Disease	Brown spots on leaf surface	High	JSON+TXT+HTML Report
Fungal Infection	White powder patches on leaf	High	Treatment Recommendation
Healthy Crop	Normal leaf texture and color	Low	Monitoring Report

where symptoms of the crop could change based on different environments.

B. Observation 2 — Weather Forecasting Supports Water Use Planning

The environmental monitoring module was able to measure environmental parameters including temperature, humidity and rainfall. The system warned of abnormal weather conditions. This is a useful tool for farmers to take action in advance, whether this means changing irrigation schedules or ensuring their crops have protection from heavy rainfall or extreme temperatures. Weather alerts can help farmers better plan their operations and minimize possible harm to crops.

C. Observation 3 — Geo-Mapping Strengthens Location-Based Monitoring

The geo-mapping module enabled the system to detect the farm location and map the farms using geographical coordinates. This aided in giving regional agricultural knowledge and weather-related information.

D. Observation 4 — Notification System Improves Farmer Response Time

The alert and notification system was helpful for disease detection and weather alerts. This indicates that real-time communication can help farmers become more aware and take prompt action on threats that might occur to their crops. The use of an integrated system to improve agricultural decision support is observed. The proposed system offers cross-platform integration of various modules in one platform. Instead of using separate instruments for crop monitoring and weather, the CropAI-SOS system has the ability to forecast, track and locate. All these functions are bundled together in one framework. The quality of detection is enhanced by additional components, interpretability and operational actionability.

TABLE X
COMPONENT-WISE IMPROVEMENT OF THE PROPOSED HYBRID PIPELINE

No.	Module	Observation	Analysis
1	Data Collection	Crop images and environmental data were successfully collected from multiple sources.	Availability of diverse data improves accuracy of crop monitoring and disease detection.
2	Data Preprocessing	Image resizing, normalization, and noise removal improved input quality.	Clean and structured data helps machine learning models perform more accurately.
3	Disease Detection Model	The machine learning model classified healthy and diseased plants.	The model demonstrates effective learning capability for plant disease identification.
4	Weather Monitoring	Temperature, humidity, and rainfall were analyzed to generate alerts.	Early weather alerts help farmers take preventive measures to protect crops.
5	Geo-Mapping Module	Farm locations were displayed using geographic coordinates.	Location-based services help provide region-specific agricultural insights.

IX. CONCLUSION

The proposed CropAI-SOS system offers an integrated solution to assist farmers in monitoring crop health and managing agricultural activities more efficiently. The system integrates artificial intelligence, weather monitoring, geo-mapping and emergency support features all in one platform. The system, incorporating these technologies, enables farmers to identify crop diseases in the early phase, get timely alerts on weather and access location-specific agricultural information.

In practice, the proposed system shows how the use of modern technologies such as machine learning and image processing can improve crop monitoring and help farmers to make better decisions. The crop disease detection module detects unhealthy crops based on image analysis. The weather alert module provides early warnings on environmental conditions that can affect crop growth. Furthermore, geo-mapping improves the system by allowing farmers to see farm locations and receive updates specific to their region.

Another important contribution of the system is the emergency support feature, which allows farmers to report urgent agricultural issues and seek assistance when needed. This transforms the platform from a simple monitoring tool into a more comprehensive agricultural support system.

X. FUTURE WORK

To further improve the proposed CropAI-SOS: AI-Powered Crop Health Monitoring, Weather Alert, Geo-Mapping, and Emergency Support System for Farmers, future work will concentrate on the following improvements:

1) **Enhanced AI capabilities:** Future enhancements could involve the incorporation of more sophisticated AI models to yield more precise diagnoses of crop diseases and tailored farming advice. Generative AI or large language

models (LLMs) could offer tailored advice for farmers on how to treat their crops, manage their soil and control pests, using real-time data.

2) **Integration of real-time data monitoring technologies:** The system can be enhanced by incorporating technologies for real-time data streaming, which continuously gather environmental and crop data from various sensors and weather services. This will enable the system to make immediate warnings to the community about weather, disease or environmental hazards to crops.
3) **Improved disease forecasting and decision making:** Future work can extend the system toward disease forecasting and improved agricultural decision support using richer historical and real-time data.
4) **Mobile Application and Farmers Accessibility:** A dedicated mobile app can be developed to ensure the system's accessibility to farmers, with emphasis on usability and ease of operation.
5) **Supervised Learning with Larger Datasets:** A larger training dataset can be used with more capable architectures. Transformers such as EfficientNet-B3/B4, ConvNeXt or Vision Transformers (ViT) could be evaluated to further enhance accuracy, and models like YOLOv8 could be considered if required for localizing the specific area (spot/lesion) of the disease.

REFERENCES

[1] S. P. Mohanty, D. P. Hughes, and M. Salathe, "Using Deep Learning for Image-Based Plant Disease Detection," *Frontiers in Plant Science*, vol. 7, 2016.
[2] K. P. Ferentinos, "Deep Learning Models for Plant Disease Detection and Diagnosis," *Computers and Electronics in Agriculture*, vol. 145, 2018, pp. 311–318.
[3] A. Kamilaris and F. X. Prenafeta-Boldu, "Deep Learning in Agriculture: A Survey," *Computers and Electronics in Agriculture*, vol. 147, 2018, pp. 70–90.
[4] K. G. Liakos et al., "Machine Learning in Agriculture: A Review," *Sensors*, vol. 18, no. 8, 2018.
[5] A. Picon et al., "Deep Convolutional Neural Networks for Mobile Capture Device-Based Crop Disease Classification," *Computers and Electronics in Agriculture*, vol. 161, 2019, pp. 280–290.
[6] S. Sladojevic et al., "Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification," *Computational Intelligence and Neuroscience*, 2016.
[7] M. Brahimi et al., "Deep Learning for Plant Diseases: Detection and Saliency Map Visualization," *Human and Machine Learning*, Springer, 2017.
[8] A. Chlingaryan, S. Sukkarieh, and B. Whelan, "Machine Learning Approaches for Crop Yield Prediction and Smart Farming," *Computers and Electronics in Agriculture*, vol. 151, 2018.
[9] S. Wolfert et al., "Big Data in Smart Farming – A Review," *Agricultural Systems*, vol. 153, 2017, pp. 69–80.
[10] P. P. Ray, "Internet of Things for Smart Agriculture: Technologies, Practices and Future Direction," *Journal of Ambient Intelligence and Smart Environments*, 2017.
[11] Q. Zhang et al., "Applications of Artificial Intelligence in Smart Agriculture: A Review," *Information Processing in Agriculture*, 2020.
[12] A. Tzounis et al., "Internet of Things in Agriculture, Recent Advances and Future Challenges," *Biosystems Engineering*, vol. 164, 2017.
[13] X. E. Pantazi et al., "Automated Leaf Disease Detection in Different Plant Species through Image Features Analysis and Machine Learning," *Computers and Electronics in Agriculture*, 2019.
[14] A. Sharma et al., "Smart Farming Using Artificial Intelligence and IoT," *International Journal of Engineering Research and Technology*, 2020.
[15] S. Ramesh et al., "Plant Disease Detection Using Deep Learning Techniques," *International Journal of Innovative Technology and Exploring Engineering*, 2019.
[16] B. Basso and J. Antle, "Digital Agriculture to Design Sustainable Agricultural Systems," *Nature Sustainability*, 2020.
[17] N. Kaur et al., "Machine Learning Applications in Precision Agriculture," *Journal of Agricultural Science and Technology*, 2021.
[18] S. S. Kamble et al., "Smart Agriculture Using Artificial Intelligence and Machine Learning," *Procedia Computer Science*, 2020.
[19] R. Singh et al., "Weather-Based Crop Monitoring System Using IoT and Machine Learning," *IEEE International Conference on Smart Computing*, 2021.
[20] H. Patel et al., "AI-Based Crop Monitoring and Disease Detection System for Smart Agriculture," *International Journal of Advanced Computer Science and Applications*, 2022.
[21] Y. LeCun, L. Bottou, Y. Bengio, and J. Haffner, "Gradient-Based Learning Applied to Document Recognition," *Proceedings of the IEEE*, vol. 86, no. 11, 1998, pp. 2278–2324.
[22] A. Krizhevsky, I. Sutskever, and E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *Advances in Neural Information Processing Systems (NeurIPS)*, 2012.
[23] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," arXiv preprint arXiv:1409.1556, 2014.
[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2016.
[25] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely Connected Convolutional Networks," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, 2017.
[26] M. Tan and Q. V. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," in *Proc. Int. Conf. Machine Learning (ICML)*, 2019.