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

Smart AI Farming: An Integrated Web-Based Decision-Support System Using Machine Learning and Deep Learning

Ms. SOPHIA LOBO
Department of Computer Science, St. Joseph's First Grade College, Mysuru, India

Abstract

Small and medium-scale farmers must make interconnected decisions about crop health, irrigation, fertilizer use, and market timing, often with limited access to specialist support or costly sensing equipment. This paper presents Smart AI Farming, a web-based agricultural decision-support prototype that brings several of these tasks together in one interface. The system combines a convolutional neural network (CNN) for leaf-image classification with image preprocessing, a recommendation component, and structured agricultural data. The application is implemented with Python and Flask; HTML, CSS, and JavaScript provide the user interface, while OpenCV prepares uploaded images for inference. The wider design also envisages irrigation guidance from soil and weather information, fertilizer suggestions from nutrient values, zone-wise advice, and crop-price forecasting. User authentication, image upload, result presentation, and basic handling of invalid inputs are included in the reported workflow. This paper presents the system as a prototype and avoids claims of measured agricultural impact or validated predictive accuracy. The work offers an integrated software architecture that can be extended through broader datasets, rigorous evaluation, and field testing.

Index Terms—Smart agriculture, precision farming, convolutional neural network, deep learning, crop disease detection, decision support, Flask, image processing.

I. INTRODUCTION

Agriculture depends on timely decisions, yet farmers may have to assess crop symptoms, determine when to irrigate, select suitable fertilizers, and decide when to sell produce under uncertain conditions. Visual inspection and experience remain important, but these approaches can be inconsistent, particularly when symptoms are difficult to distinguish or when several factors affect a crop at once. Advanced precision-agriculture systems can add sensors, drones, and automated control; however, these technologies may also introduce equipment, connectivity, and maintenance costs.

Artificial intelligence (AI) provides tools for supporting selected agricultural decisions. Machine-learning methods can learn patterns from structured data, while deep-learning models—especially CNNs—can extract visual features from plant-leaf images. In practice, however, separate tools for disease identification, irrigation, fertilizer planning, and market information may be difficult for users to navigate. An accessible interface that organizes these functions can help make digital decision support more practical.

This paper presents Smart AI Farming, a software-based prototype that combines image-based crop disease prediction with a decision engine and structured agricultural information. The primary contribution is the integration of these functions within a modular web application rather than a claim of a new neural-network architecture. The paper describes the system design and implementation, summarizes the reported tests, and identifies the evidence needed before the prototype can be considered a validated tool for farm use.

II. RELATED WORK

Research in smart agriculture spans IoT-enabled irrigation, image-based disease recognition, robotics, and data-driven crop management. The paper reviews work on IoT and cloud-supported irrigation, including systems that use soil-moisture and environmental sensors to support water management [1], [3]. These approaches demonstrate the value of timely field data, while also highlighting practical dependencies such as sensor upkeep, connectivity, and computing resources.

For crop disease recognition, recent reviews examine conventional machine-learning methods and deep-learning architectures, including CNN-based image classifiers and object-detection approaches [2], [7]. Their reported limitations include dataset imbalance, changes in performance between curated images and field conditions, and computational requirements. Broader reviews of smart farming discuss the integration of AI, IoT, unmanned aerial vehicles, and robotics, while noting costs and deployment barriers [4]–[6], [8]. A review of deep learning for crop prediction is also cited in the source report [9].

The system described here responds to a practical integration gap by placing image-based classification and additional decision-support modules behind one web interface. Unlike sensor-intensive deployments, the core disease-detection workflow is based on a user-uploaded image. This lowers the hardware requirements of the interaction model, although the quality of recommendations still depends on the quality, coverage, and freshness of the underlying data.

III. SYSTEM OVERVIEW AND DESIGN

The application follows a layered architecture with five logical parts: user interaction, frontend, backend, AI model, and data storage. A farmer or other user accesses the application through a browser, signs in, and uploads a leaf image. The frontend sends the request to a Flask backend, which validates the upload, preprocesses the image, invokes the trained model, and passes the prediction to a decision component. The response is then rendered on the result page.

The frontend uses HTML, CSS, and JavaScript for page structure, presentation, and interaction. Flask handles routes, request processing, template rendering, and session state. It identifies TensorFlow/Keras as the deep-learning framework and OpenCV and NumPy as image-processing tools. Structured recommendation information is read from CSV files; the implementation listing also includes SQLite tables for user accounts and prediction history. This distinction is important: the report describes more than one data-storage mechanism, with CSV files used for recommendation data and SQLite used for application records.

The system is designed to reject missing or unsuitable uploads and to display a warning when the prediction function indicates an unclear image, a missing leaf, or an error. The paper also describes a decision engine that can retrieve fertilizer, irrigation, weather, and market-related information.

IV. METHODOLOGY

A. Image Input and Preprocessing

The user submits a leaf image through the web interface. The backend checks that an image has been provided and that the upload is not empty. OpenCV is used to read the image, convert its color representation from BGR to RGB, and resize it to 224 × 224 pixels so that the input dimensions match the model's expected format.

B. CNN-Based Classification

The disease-identification component is described as a trained CNN implemented with TensorFlow/Keras. Its processing flow consists of convolutional filters for feature extraction, nonlinear activation, pooling, flattening, fully connected classification, and an output layer. The model returns class probabilities, from which the application selects a predicted crop or disease class and displays a confidence value. The trained model is saved in an H5 file and loaded when the application runs.

C. Decision Support

After classification, a decision engine uses the prediction to retrieve related information from structured data. The fertilizer suggestions and treatment or preventive guidance, and the broader system design includes irrigation requirements, weather information, zone-wise recommendations, and crop-price forecasts. Irrigation guidance is intended to use soil and weather variables; fertilizer recommendations are intended to consider soil NPK values; and price forecasts are intended to cover a six-month horizon.

D. Web Workflow and User Records

Flask routes support registration, login, dashboard access, image upload, prediction, result display, and profile/history access. The code listing shows SQLite tables for users and prediction history, including crop, disease, confidence, image path, and timestamp fields. CSV-based files are used for selected recommendation data. Authentication and session management are included in the intended workflow.

V. IMPLEMENTATION

The prototype is implemented primarily in Python. Flask coordinates requests and model inference, while TensorFlow/Keras provides the CNN runtime. OpenCV and NumPy handle image preparation, and HTML/CSS/JavaScript form the browser interface. The reported minimum development environment is a computer with an Intel i3/i5-class processor, 4 GB RAM, and 100 GB of disk space, running Windows or Linux. These are reported development requirements, not a benchmark of production-scale capacity.

The interface includes welcome, login, dashboard, upload, result, details, and profile pages. The history feature is intended to let a user revisit recent predictions. This design separates the interface, inference function, and decision engine, allowing individual components to be revised without redesigning the entire application. The modular structure is useful for future extension, but scalability and concurrent-user performance have not been quantified in the report.

VI. TESTING AND OBSERVATIONS

The report lists unit, integration, system, functional, usability, and performance testing as relevant testing activities. Its test-case table records successful outcomes for image upload, invalid-image handling, preprocessing,

model prediction, confidence display, recommendation display, result-page rendering, and navigation. The evidence is limited in scope.

VII. DISCUSSION

The main strength of Smart AI Farming is its integrated workflow: users can submit a leaf image, receive a model-generated classification, and view related information through one browser-based application. Its software-first design avoids making dedicated sensors or drones mandatory for the basic image-classification path. The layered architecture also provides a clear route for adding data sources and improving individual modules.

Several limitations remain. First, the report does not document

VIII. CONCLUSION AND FUTURE WORK

This paper has described Smart AI Farming, a web-based prototype that combines CNN-based leaf-image classification with a modular decision-support workflow. The implementation uses Flask, TensorFlow/Keras, OpenCV, and structured data sources to connect image upload, preprocessing, prediction, result display, and recommendations. The paper successful functional test cases for the principal application flow.

Further extensions may incorporate IoT sensors, drone or satellite imagery, expert advisory services, and links to public agricultural schemes. Security hardening, usability studies with intended users, and controlled field trials are necessary steps toward a dependable deployment.

REFERENCES

[1] A. Morchid, H. Qjidaa, R. E. Alami, S. Mobayen, P. Skruch, and B. Bossoufi, “Smart irrigation-based internet of things and cloud computing technologies for sustainable farming,” Scientific Reports, 2026. [Bibliographic details should be verified.]

[2] T. Murugan et al., “Research advances in maize crop disease detection using machine learning and deep learning approaches,” Computers, vol. 15, no. 2, Art. no. 99, 2026. [Verify author list and publication metadata.]

[3] A. Y. B. Ahmad, J. A. Alzubi, C. K., S. Urooj, and M. Shahzad, “Automated smart drip irrigation system in internet of things using adaptive residual hybrid network for precision farming,” Scientific Reports, vol. 16, no. 1, Art. no. 4327, 2026. [Verify author initials and metadata.]

[4] M. Raj and M. Prahadeeswaran, “Revolutionizing agriculture: A review of smart farming technologies for a sustainable future,” Discover Applied Sciences, vol. 7, no. 9, Art. no. 937, 2025.

[5] S. Pearson et al., “Robotics and autonomous systems for net zero agriculture,” Current Robotics Reports, vol. 3, no. 2, pp. 57–64, 2022.

[6] M. Dutta et al., “Internet of things-based smart precision farming in soilless agriculture: Opportunities and challenges for global food security,” IEEE Access, 2025. [Add volume, pages/article number, and DOI if available.]

[7] A. Kaur, G. S. Randhawa, F. Abbas, M. Ali, T. J. Esau, A. A. Farooque, and R. Singh, “Artificial intelligence driven smart farming for accurate detection of potato diseases: A systematic review,” IEEE Access, vol. 12, pp. 193902–193922, 2024.

[8] D. F. Yépez-Ponce, J. V. Salcedo, P. D. Rosero-Montalvo, and J. Sanchis, “Mobile robotics in smart farming: Current trends and applications,” Frontiers in Artificial Intelligence, vol. 6, Art. no. 1213330, 2023.

[9] A. Oikonomidis, C. Catal, and A. Kassahun, “Deep learning for crop yield prediction: A systematic literature review,” New Zealand Journal of Crop and Horticultural Science, 2022. [Verify title and full publication metadata against the original source.]