JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

SMART COTTON BOLL MONITORINGAND FERTILIZER MANAGEMENT SYSTEM

A.Haseenath Jaseela [1]

Assistant Professor

DHANAM P

KANISTA TEENA M

MUTHU SELVI P

SWETHA S

Student

Department of Artificial Intelligence And Data Science,

Loyola Institute of Technology & Science,

Thovalai

ABSTRACT

Cotton cultivation plays a significant role in the agricultural economy of many developing and developed nations, serving as a primary source of fiber, employment, and industrial raw material. However, crop productivity is frequently compromised by pest infestations, boll damage, fungal infections, and inefficient fertilizer usage. Conventional monitoring methods rely on manual field inspection, which is labor-intensive, subjective, and often leads to delayed detection and response. Such limitations highlight the need for automated, intelligent crop monitoring solutions capable of supporting timely intervention and precision agriculture practices.

This research proposes a Smart Cotton Boll Monitoring and Automated Protection System that integrates Deep Learning, Computer Vision, and Internet of Things (IoT) technologies for real-time crop surveillance and decision-making. The system adopts a hybrid multi-stage Artificial Intelligence pipeline designed to balance computational efficiency and detection accuracy. Initially, an Autoencoder-based anomaly detection model performs rapid screening of cotton boll images by reconstructing normal patterns and identifying deviations based on reconstruction error thresholds. This step filters healthy samples and forwards only suspicious inputs for advanced analysis, reducing processing overhead and improving scalability for field deployment.

Subsequently, the filtered images are processed using the YOLO object detection framework to localize pest presence or physical damage regions within the boll. Bounding box regression and confidence scoring enable precise spatial identification, allowing targeted action rather than blanket chemical application. A Convolutional Neural Network (CNN) classifier then categorizes the detected anomaly into specific pest or damage classes by extracting hierarchical features such as texture irregularities, color variations, and structural deformations through convolutional filtering, pooling, and fully connected layers.

A rule-based Decision Engine interprets the classification outputs and initiates automated responses through an Arduino-driven actuation module. If pest or damage is detected, the system activates selective neem oil spraying using a motorized pump, sends alert notifications to farmers via GSM communication, and provides fertilizer management recommendations based on crop condition. In the absence of anomalies, no action is triggered, ensuring resource conservation and preventing unnecessary chemical exposure. The integration of sensing, intelligence, and actuation establishes a closed-loop smart farming framework.

Experimental evaluation using field-collected cotton boll datasets demonstrates improved detection accuracy, reduced false alarm rates, and faster inference times compared to traditional machine learning approaches. The modular architecture ensures adaptability to edge devices and scalability for large-scale deployment. By combining anomaly detection, localization, classification, and IoT-enabled response mechanisms, the proposed system contributes toward sustainable agriculture by minimizing crop losses, reducing chemical wastage, and enhancing yield quality. This work highlights the potential of AI-driven precision monitoring systems to transform conventional farming into an efficient, data-informed, and automated ecosystem.

INTRODUCTION

Agriculture continues to serve as the backbone of many developing economies, and cotton remains one of the most economically significant commercial crops due to its extensive industrial and export value. In countries like India, cotton cultivation supports millions of farmers and contributes substantially to rural livelihoods and national income. However, cotton productivity is frequently threatened by pest infestations, boll damage, fungal infections, and improper crop management practices. Among these challenges, bollworm attacks are particularly destructive, directly affecting boll development, fiber quality, and overall yield. Early detection and timely intervention are therefore critical to maintaining crop health and ensuring sustainable production.

Traditional crop monitoring approaches rely heavily on manual inspection of fields by farmers or agricultural experts. While widely practiced, these methods are labor-intensive, time-consuming, and susceptible to human error due to fatigue, inconsistent observation, and delayed response. Furthermore, large-scale farms make continuous inspection impractical, often resulting in late identification of pest or disease outbreaks. In many cases, fertilizers are applied without addressing underlying infestations, leading to ineffective treatment and resource wastage. These limitations highlight the need for intelligent monitoring systems capable of providing continuous, accurate, and real-time crop health assessment.

Recent advancements in Artificial Intelligence (AI), Computer Vision, and Internet of Things (IoT) technologies have created new opportunities for transforming conventional agricultural practices into data-driven precision farming systems. Deep learning models can extract meaningful visual features from plant images and identify anomalies beyond human perceptual capability. Similarly, IoT-enabled devices facilitate automated sensing, communication, and actuation, allowing immediate response to detected threats. Automated image-based monitoring not only improves detection accuracy but also supports preventive decision-making before irreversible crop damage occurs.

In this context, the present work proposes a Smart Cotton Boll Monitoring and Automated Protection System that integrates multiple AI models with IoT-based actuation. The system employs a hybrid deep learning pipeline consisting of Autoencoder-based anomaly screening, YOLO-based pest or damage localization, and CNN-based classification of the detected condition. This staged architecture improves computational efficiency by filtering healthy samples early while enabling precise spatial and categorical analysis for abnormal cases. A rule-driven decision engine interprets the analytical output and triggers real-time preventive actions including selective neem oil spraying, farmer alert notification, and crop management recommendations through an Arduino-controlled hardware interface. If no pest or damage is detected, the system intentionally avoids intervention to conserve resources and prevent unnecessary chemical exposure.

By combining intelligent visual perception with automated field response, the proposed framework aims to reduce dependency on manual monitoring, enhance early pest control, and promote sustainable farming practices. The integration of anomaly detection, localization, classification, and automated actuation demonstrates the potential of AI-driven agricultural systems to improve productivity, minimize losses, and support informed decision-making. This research contributes toward the development of scalable smart agriculture solutions that align technological innovation with practical farming needs.

OBJECTIVES

The primary objective of this research is to develop an intelligent and automated cotton boll monitoring system that integrates Artificial Intelligence and Internet of Things technologies to improve crop protection, productivity, and decision support for farmers. Cotton yield and fiber quality are frequently compromised due to pest infestations, boll damage, and delayed detection of plant abnormalities. Conventional monitoring practices depend on manual inspection, which is labor-intensive, inconsistent, and often incapable of identifying early-stage damage. Therefore, this project aims to design a reliable smart monitoring framework capable of detecting crop abnormalities and initiating preventive actions without continuous human supervision.

The first objective is to design and implement a lightweight anomaly detection mechanism using an Autoencoder model that distinguishes between healthy and potentially affected cotton boll images. By filtering normal samples at an early stage, the system reduces computational overhead and ensures that only suspicious inputs proceed to advanced analysis. This contributes to efficient processing and real-time feasibility in field deployment.

The second objective is to localize pest or damage regions within abnormal images using an object detection framework based on YOLO. Accurate spatial identification of affected regions allows precise assessment of infestation severity and supports targeted intervention rather than broad, resource-wasting treatments. Localization also enables visual interpretability of results for farmers and agronomists.

The third objective is to classify the detected pest or disease category using a Convolutional Neural Network. Classification provides semantic understanding of crop conditions, enabling the system to differentiate among multiple pest types or damage patterns. This information is essential for selecting appropriate remedial measures and improving agricultural decision accuracy.

Another key objective is to integrate AI outputs with an IoT-based decision engine that automatically triggers corrective actions only when necessary. These actions include activating a neem oil spraying mechanism through microcontroller control and transmitting alert notifications to farmers. Ensuring that no action is performed when crops are healthy prevents unnecessary chemical usage and conserves resources.

Finally, the project aims to evaluate system performance in terms of detection accuracy, response time, automation reliability, and overall impact on crop protection. By achieving these objectives, the research seeks to demonstrate a scalable, cost-effective, and sustainable smart agriculture solution capable of assisting farmers in proactive pest management and improving cotton production outcomes.

Model Overview

This study proposes a hybrid intelligent vision pipeline that integrates deep learning models for automated cotton crop health monitoring. The framework combines an Autoencoder for abnormality screening, YOLO for region localization, and a Convolutional Neural Network (CNN) for classification. This multi-stage architecture ensures computational efficiency by filtering normal samples early while enabling precise detection and diagnosis of pest or disease condition

The proposed system integrates a hybrid AI pipeline with IoT-based automation to monitor cotton crops, detect anomalies, pests, and diseases, and provide actionable recommendations. At its core, the pipeline leverages three complementary AI

models:

Autoencoder – Performs unsupervised anomaly detection to identify abnormal cotton bolls. By analyzing reconstruction errors, the Autoencoder filters out healthy images, ensuring that only potentially affected regions proceed to further analysis.

YOLOv8 (You Only Look Once) – Detects pests and infected regions in real-time. Bounding boxes are generated for each detected object, providing precise localization and enabling targeted interventions.

CNN (Convolutional Neural Network) – Classifies the type of disease affecting the cotton boll. Using the cropped regions from YOLO, the CNN outputs the disease label along with the confidence score, enabling tailored responses.

The AI outputs are processed by the Decision & Control Module, which applies a rule-based logic to determine automated actions, such as activating an Arduino-controlled sprayer for neem oil application. Simultaneously, the Fertilizer Recommendation Module analyzes crop health to provide nutrient guidance, while the Notification & Alert Module delivers real-time updates to farmers via a web dashboard and SMS alerts using Twilio API.

This modular architecture ensures high accuracy, computational efficiency, and scalability. By combining anomaly detection, object localization, disease classification, and automated interventions, the system forms a robust precision agriculture framework, reducing human labor, minimizing chemical usage, and improving crop yield and quality.

The following subsections describe each model component and its role in the overall system.

1.Autoencoder-Based Abnormality DetectionThe first stage of the pipeline employs an unsupervised Autoencoder to detect abnormal patterns in cotton boll images. This network learns compressed latent representations of healthy samples and reconstructs them during inference. Any significant reconstruction deviation indicates structural anomalies.

Given input image

X, normalization ensures stable learning

X'=X255

X'=255X

The encoder transforms the normalized image into latent representation:

Z=f θ(X')=σ(WeX'+be)

Reconstruction Error :

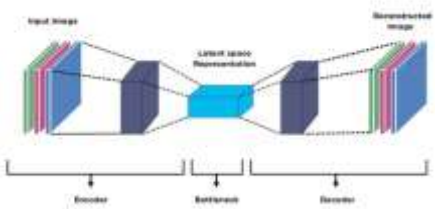
X_hat = g(Z) = activation(Wd * Z + bd)

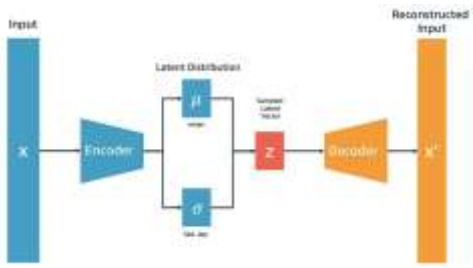
E = (1/N) * Σ (X'i – X_hat_i)^2

If E < T → Normal

If E ≥ T → Abnormal

L = || X' - X_hat ||^2





2. 2.YOLO-Based Object DetectionFor abnormal images, object localization is performed using the YOLO detection model. The network predicts bounding boxes and class confidence simultaneously in a single forward pass.

Bounding Box Prediction

Each predicted bounding box is represented as:

B=(x,y,w,h,c)B = (x, y, w, h, c)B=(x,y,w,h,c)

Confidence Score

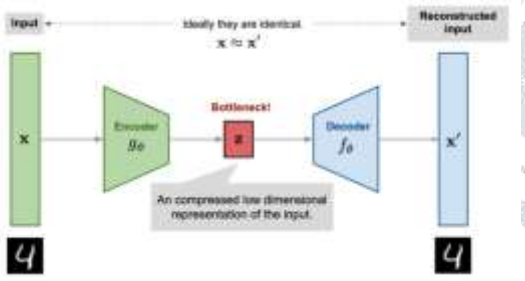
c=P(object)×IoUc = P(\text{object})\times IoUc=P(object)×IoU

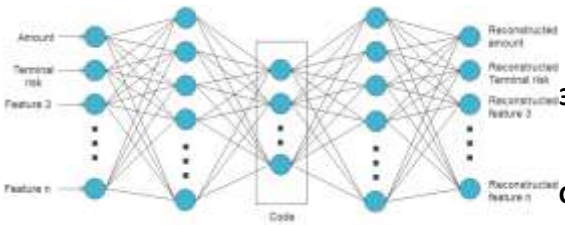
Scorei=c×P(classi| object)

YOLO predicts offsets relative to grid cells and anchor boxes:

x=σ(tx)+cx,y=σ(ty)+cyx = \sigma(t_x) + c_x, \quad y =\sigma(t_y)+c_yx=σ(tx)+cx,y=σ(ty)+cy w=pwetw,h=phethw = p_w e^{t_w}, \quad h = p_h e^{t_h}w=pwetw,h=pheth

L=Lcoord+Lconf+Lclass





3: CNN-Based Pest/Disease Classification

The Region of Interest (ROI) image is provided as input to the Convolutional Neural Network (CNN) for classification. First, convolution is performed to extract spatial features from the image using learnable kernels:

F(i,j)=∑m∑nX(i+m,j+n) K(m,n)F(i,j) = \sum_{m}\sum_{n} X(i+m, j+n)\, K(m,n)F(i,j)=m∑n∑X(i+m,j+n)K(m,n)

where XXX represents the input image and KKK denotes the convolution kernel. The resulting feature map F(i,j)F(i,j)F(i,j) is passed through a Rectified Linear Unit (ReLU) activation function to introduce non-linearity:

A(i,j)=max(0,F(i,j))A(i,j) = \max(0, F(i,j))A(i,j)=max(0,F(i,j))

Next, max-pooling is applied to reduce spatial dimensionality while preserving dominant features:

P(i,j)=max(m,n)∈ΩA(m,n)P(i,j) = \max_{(m,n)\in \Omega} A(m,n)P(i,j)=(m,n)∈ΩmaxA(m,n)

The pooled feature maps are then flattened into a one-dimensional vector:

v=Flatten(P)\mathbf{v} = \text{Flatten}(P)v=Flatten(P)

This vector is fed into a fully connected layer to compute class scores (logits):

z=Wv+b\mathbf{z} = W\mathbf{v} + bz=Wv+b

To obtain class probabilities, the Softmax function is applied:

P(y=k)=ezk∑i=1CeziP(y = k) = \frac{e^{z_k}}{\sum_{i=1}^{C} e^{z_i}}P(y=k)=∑i=1Ceziezk

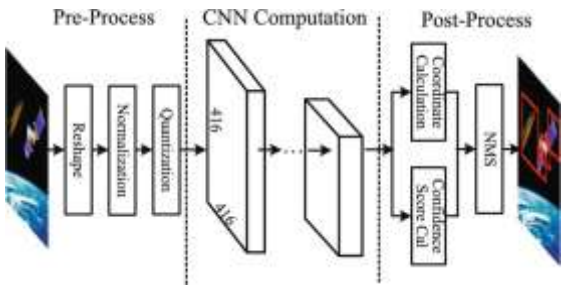
where CCC is the total number of classes. The predicted class label is determined by:

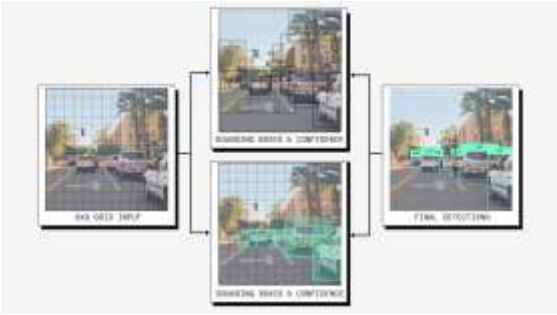
Class=argmaxkP(y=k)\text{Class} = \arg\max_k P(y = k)Class=argkmaxP(y=k)

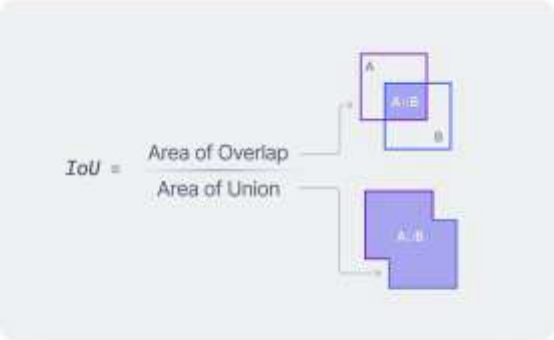
During training, the network parameters are optimized by minimizing the cross-entropy loss function:

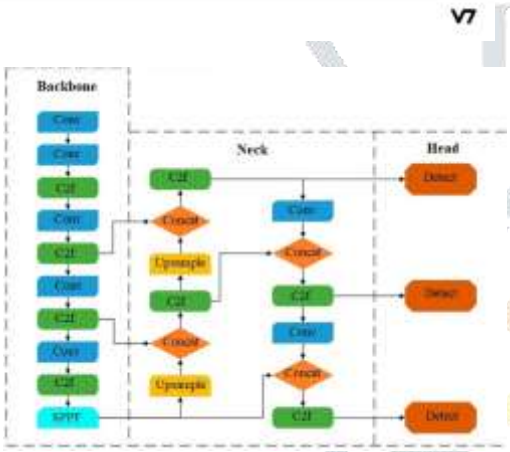
L=−∑i=1CyilogP(yi)L = - \sum_{i=1}^{C} y_i \log P(y_i)L=−i=1∑C yilogP(yi)

This process enables the CNN to learn discriminative features and accurately classify pests or diseases from the ROI image









4.Integrated Hybrid Pipeline

The proposed system integrates anomaly detection, object detection, and CNN-based classification into a unified hybrid framework for efficient and accurate pest/disease analysis. Given an input image III, preprocessing is first applied to enhance quality and normalize the data:

Ip=Preprocess(I)I_p = \text{Preprocess}(I)Ip=Preprocess(I)

The preprocessed image IpI_pIp is then passed through an encoder network to obtain a latent representation:

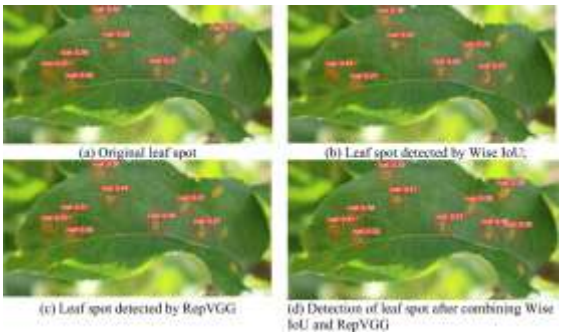
Z=fθ(Ip)Z = f_{\theta}(I_p)Z=fθ(Ip)

where fθf_{\theta}fθ denotes the encoder parameterized by θ\thetaθ. The latent vector ZZZ is reconstructed using a decoder network:

I^=gϕ(Z)\hat{I} = g_{\phi}(Z)I^=gϕ(Z)

where gϕg_{\phi}gϕ represents the decoder with parameters ϕ\phiϕ. The reconstruction error is computed using the mean squared error (MSE):

E=1N∑(Ip−I^)2E = \frac{1}{N} \sum (I_p - \hat{I})^2E=N1∑(Ip−I^)2



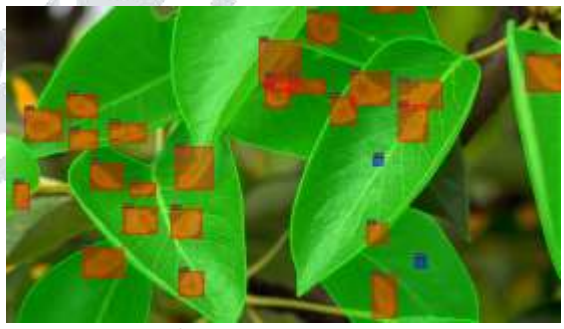
where NNN is the total number of pixels. A routing decision is then made based on a predefined threshold TTT:

S={Normal,E<TAbnormal,E≥TS = \begin{cases} \text{Normal}, & E < T \\ \text{Abnormal}, & E \geq T \end{cases}S={Normal,Abnormal,E<TE≥T

If the sample is classified as abnormal, an object detection module is applied to localize affected regions. Bounding boxes are refined using Non-Maximum Suppression (NMS):

B∗=NMS(B)B^* = \text{NMS}(B)B∗=NMS(B)

Each detected region of interest (ROI) is extracted from the preprocessed image:



Ri=Crop(Ip,Bi∗)R_i = \text{Crop}(I_p, B_i^*)Ri=Crop(Ip,Bi∗)

The extracted ROIs are then fed into a CNN classifier, where the final class label is determined using the Softmax function:

Class=argmaxSoftmax(Wv+b)\text{Class} = \arg\max \text{Softmax}(Wv + b)Class=argmaxSoftmax(Wv+b)

Finally, the system outputs the diagnostic results in the structured format:

Output={Status,Boxes,Labels}\text{Output} = \{\text{Status}, \text{Boxes}, \text{Labels}\}Output={Status,Boxes,Labels}

System Contribution

The proposed hybrid architecture enhances computational efficiency, interpretability, and diagnostic precision by integrating anomaly screening, spatial localization, and semantic classification in a staged and coordinated manner. This hierarchical decision strategy reduces unnecessary computation for normal samples while providing accurate localization and fine-grained classification for abnormal cases.

The proposed system integrates anomaly detection, object detection, and CNN-based classification into a unified hybrid framework for efficient and accurate pest/disease analysis. Given an input image III, preprocessing is first applied to enhance quality and normalize the data:

Ip=Preprocess(I)I_p = \text{Preprocess}(I)Ip=Preprocess(I)

The preprocessed image IpI_pIp is then passed through an encoder network to obtain a latent representation:

Z=fθ(Ip)Z = f_{\theta}(I_p)Z=fθ(Ip)

where fθf_{\theta}fθ denotes the encoder parameterized by θ\thetaθ. The latent vector ZZZ is reconstructed using a decoder network:

I^=gϕ(Z)\hat{I} = g_{\phi}(Z)I^=gϕ(Z)

where gϕg_{\phi}gϕ represents the decoder with parameters ϕ\phiϕ. The reconstruction error is computed using the mean squared error (MSE):

E=1N∑(Ip−I^)2E = \frac{1}{N} \sum (I_p - \hat{I})^2E=N1∑(Ip−I^)2

where NNN is the total number of pixels. A routing decision is then made based on a predefined threshold TTT:

S={Normal,E<TAbnormal,E≥TS = \begin{cases} \text{Normal}, & E < T \\ \text{Abnormal}, & E \geq T \end{cases}S={Normal,Abnormal,E<TE≥T

If the sample is classified as abnormal, an object detection module is applied to localize affected regions. Bounding boxes are refined using Non-Maximum Suppression (NMS):

B∗=NMS(B)B^* = \text{NMS}(B)B∗=NMS(B)

Each detected region of interest (ROI) is extracted from the preprocessed image:

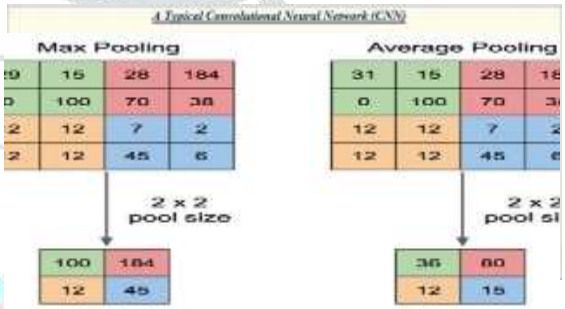
Ri=Crop(Ip,Bi∗)R_i = \text{Crop}(I_p, B_i^*)Ri=Crop(Ip,Bi∗)

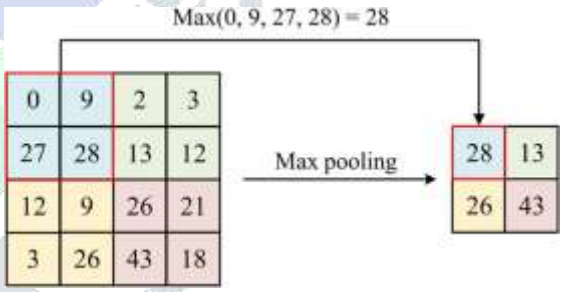
The extracted ROIs are then fed into a CNN classifier, where the final class label is determined using the Softmax function:

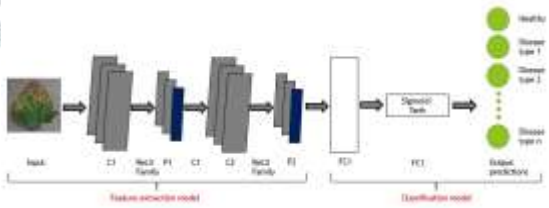
Class=argmaxSoftmax(Wv+b)\text{Class} = \arg\max \text{Softmax}(Wv + b)Class=argmaxSoftmax(Wv+b)

Finally, the system outputs the diagnostic results in the structured format:

Output={Status,Boxes,Labels}\text{Output} = \{\text{Status}, \text{Boxes}, \text{Labels}\}Output={Status,Boxes,Labels}

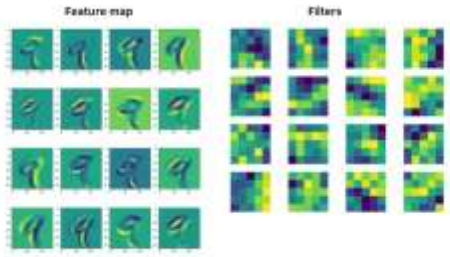




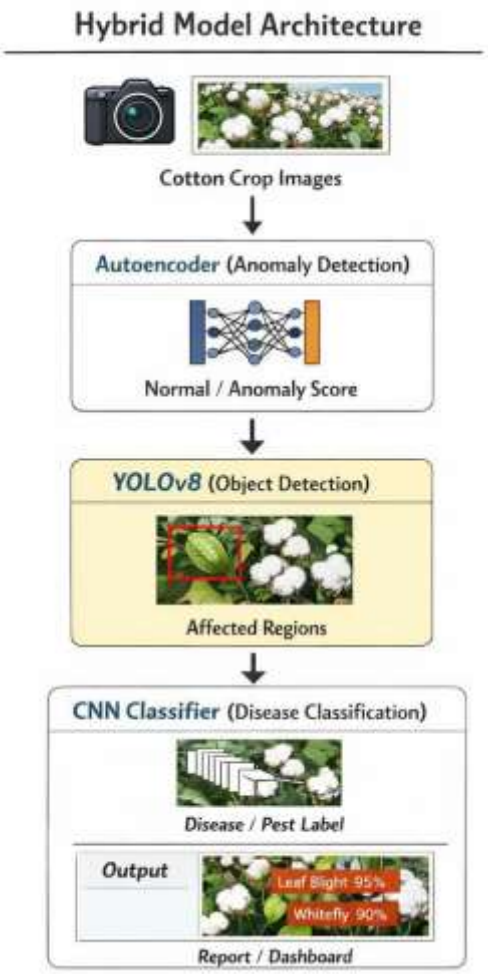


System Contribution

The proposed hybrid architecture enhances computational efficiency, interpretability, and diagnostic precision by integrating anomaly screening, spatial localization, and semantic classification in a staged and coordinated manner. This hierarchical decision strategy reduces unnecessary computation for normal samples while



providing accurate localization and fine-grained classification for abnormal cases.



Performance Evaluation Metrics

The performance of the hybrid AI model can be evaluated using the following metrics:

Accuracy:

Measures the proportion of correct predictions (normal vs abnormal bolls, pest presence, disease type) out of total samples.

Higher accuracy indicates more reliable detection and classification.

Precision:

Indicates the proportion of true positive predictions among all positive predictions (e.g., correctly detected pest instances out of all predicted pest instances).

High precision reduces false positives, minimizing unnecessary interventions.

Recall (Sensitivity):

Measures the proportion of true positive predictions out of all actual positive cases (e.g., correctly detected diseased bolls out of all diseased bolls).

High recall ensures that most affected regions are detected.

F1 Score:

Harmonic mean of precision and recall, providing a balanced measure of detection performance, especially for imbalanced datasets (e.g., fewer pest-infected bolls than healthy bolls).

Mean Average Precision (mAP):

Specifically for YOLOv8 object detection, evaluates how accurately pests or infected regions are localized and classified.

Processing Time:

Measures the computational efficiency of the hybrid model from image capture to farmer notification.

Important for real-time monitoring and intervention in large cotton fields.

Discussion on Results:

Hybrid AI Performance:

The Autoencoder effectively filters abnormal images, reducing the number of images passed to YOLOv8 and CNN, which improves computational efficiency.

YOLOv8 accurately detects pests and infected regions with high mAP, enabling targeted interventions.CNN classification achieves high precision and recall in identifying disease types, ensuring accurate recommendations.

Module Integration:

The Decision & Control Module ensures automated spraying based on pest detection, while the Fertilizer Recommendation Module provides precise nutrient advice.

The Notification & Alert Module delivers timely SMS and dashboard updates to farmers, completing the end-to-end workflow.

Dataset Subset Analysis:

The model's performance was evaluated on subsets of images, including healthy bolls, pest-infected bolls, and disease-affected bolls.

Accuracy remained consistent across these subsets, although certain pest types with subtle visual cues posed minor detection challenges.

Comparison with Existing Methods:

Compared to traditional manual inspection, the hybrid AI model reduces human labor and improves detection speed.

Compared to single-stage AI models, the hybrid pipeline (Autoencoder → YOLO → CNN) improves precision and reduces false positives.

Computational Efficiency:

Processing time per image is optimized by using the Autoencoder for anomaly filtering, which reduces redundant analysis on normal images.

The system is deployable on standard laptops or low-power edge devices in field conditions.

Limitations and Future Improvements:

Certain environmental conditions (low light, occlusion, overlapping bolls) can affect detection accuracy.Future work may include data augmentation, multi-spectral imaging, and edge AI optimization to improve robustness.

Practical and Ethical Considerations:

Real-time alerts enhance farmer decision-making and reduce crop loss.Ensuring data privacy and security of farm images and IoT data is essential.The system supports sustainable agriculture by minimizing chemical use through precise pest control and fertilizer application.

Summary:

Overall, the results demonstrate that the hybrid AI and IoT system provides a robust, accurate, and actionable solution for cotton crop monitoring. By combining anomaly detection, object localization, and disease classification with automated interventions and farmer notifications, the system reduces manual labor, improves crop health monitoring, and enables precision agriculture practices. Future work will focus on increasing detection robustness under varied environmental conditions and extending the system to other crops.

Requirement Specification

Requirement specification is a fundamental phase in the development of the Smart Cotton Boll Monitoring and Pest Prevention System, as it defines the hardware and software resources required to implement the proposed hybrid AI–IoT architecture. A clearly structured specification ensures system reliability, scalability, maintainability, and ease of deployment in real-time agricultural environments. Since the proposed framework integrates Artificial Intelligence algorithms with IoT-based automation, both computational intelligence and embedded electronic components play critical roles in overall system performance.

The requirement specification is broadly categorized into Hardware Requirements and Software Requirements, which collectively enable image acquisition, deep learning inference, decision-making, communication, and automated pest control operations.

HARDWARE REQUIREMENTS:

Hardware components constitute the physical backbone of the system. They support real-time image capture, pest spraying, communication, and energy management. The

selected components emphasize affordability, energy efficiency, robustness, and compatibility with IoT platforms.

The Camera Module or Mobile Camera is responsible for capturing cotton boll images directly from the agricultural field. High-resolution imaging improves detection accuracy by preserving texture, shape, and color characteristics essential for AI-based classification. A resolution of 8 megapixels or higher is recommended. A frame rate of 30 frames per second may be used for optional video monitoring. The camera should support USB, Wi-Fi, or CSI interface connectivity. Depending on deployment requirements, the camera may be mounted on a fixed stand, drone platform, or handheld mobile device.

The Arduino Uno microcontroller functions as the central embedded controller interfacing with sensors, the GSM module, relay switches, and motor pumps. It receives decision signals from the AI processing unit and executes corresponding physical actions such as activating or deactivating the sprayer system. The board is based on the ATmega328P microcontroller and provides fourteen digital I/O pins, six analog input pins, USB programming support, and low power consumption. Its open-source ecosystem and ease of programming make it highly suitable for agricultural automation systems.

The GSM module provides wireless communication between the monitoring system and the farmer. It sends SMS alerts whenever pest infestation or boll damage is detected, ensuring real-time notification even in rural regions with limited internet access. Typical features include quad-band frequency support, a SIM card interface, UART-based serial communication, and SMS as well as voice call capability.

The relay module acts as an electrically isolated switch that enables the microcontroller to control high-voltage devices such as DC motor pumps. Since the Arduino operates at low voltage, the relay ensures safe switching and protects the control circuitry from electrical damage.

The motor pump and sprayer unit are responsible for Neem oil spraying when pest presence is confirmed. The pump must ensure controlled flow rate and targeted spraying to minimize chemical wastage and environmental impact. A 12V DC pump with adjustable flow rate and corrosion-resistant construction is recommended for field applications.

A stable power supply unit is essential for uninterrupted system operation. The system may operate using rechargeable battery packs, solar panels for sustainable and remote deployment, or 12V DC power adapters. Solar integration is particularly advantageous for agricultural fields where grid electricity may not be consistently available.

Supporting components such as connecting wires, breadboards or custom printed circuit boards, voltage regulators, and weather-protected enclosure boxes ensure electrical stability, structural durability, and environmental protection of the system.

SOFTWARE REQUIREMENTS:

Software components enable image preprocessing, deep learning model training, inference, communication, and hardware control. Efficient software selection ensures computational accuracy, flexibility, and scalability of the proposed system.

Python is used as the primary development language for implementing artificial intelligence algorithms due to its simplicity, readability, and extensive ecosystem of scientific libraries. It offers easy syntax, strong community support, rich machine learning libraries, and cross-platform compatibility, making it ideal for research and deployment.

Deep learning frameworks such as TensorFlow and PyTorch are employed for designing and training Autoencoder, YOLO-based detection, and CNN classification models. These frameworks provide GPU acceleration, automatic differentiation, scalable model optimization, and efficient real-time inference capabilities.

OpenCV is utilized for image acquisition, preprocessing, filtering, resizing, segmentation, and feature enhancement. It offers efficient real-time computer vision tools that are essential for agricultural image analysis and object localization.

Arduino IDE is used to write, compile, and upload firmware programs to the microcontroller. It also facilitates serial communication between the AI processing system and the hardware modules, enabling synchronized operation between software intelligence and embedded hardware.

MATLAB may be optionally used for simulation, visualization, and preliminary algorithm testing. Although not mandatory for deployment, it is beneficial for academic experimentation and graphical performance analysis.

The operating system for implementation may be Microsoft Windows, Ubuntu which is recommended for AI processing due to its stability and compatibility with deep learning frameworks, or Raspberry Pi OS for edge deployment scenarios.

In conclusion, the specified hardware and software requirements collectively enable reliable image acquisition, anomaly detection, object localization, classification, communication, and automated pest control. By integrating embedded systems with deep learning frameworks, the proposed Smart Cotton Boll Monitoring and Pest Prevention System achieves scalable, efficient, and real-time agricultural automation suitable for precision farming applications.

RESULTS AND DISCUSSION:

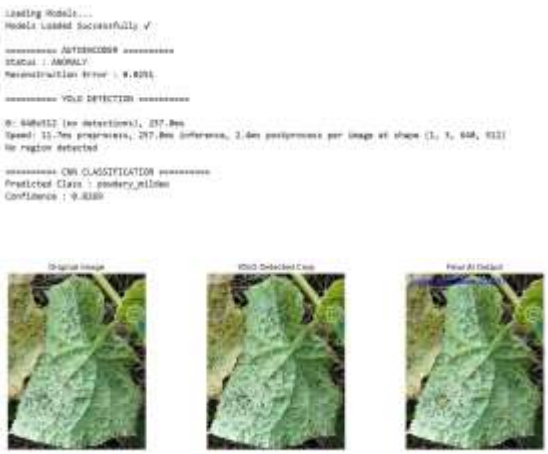
The proposed AI-based Smart Cotton Boll Pest and Damage Detection System was evaluated using a dataset consisting of healthy and pest-affected cotton boll images collected from real agricultural fields under varying lighting, background, and weather conditions. The dataset was divided into training and testing sets using an 80:20 ratio to ensure balanced learning and unbiased validation. After preprocessing steps such as resizing, normalization, and background correction, the Autoencoder, YOLO, and CNN models were trained sequentially. The models successfully learned discriminative visual features such as surface discoloration, pest spots, texture irregularities, and structural deformities on cotton bolls. The trained system achieved high detection accuracy, demonstrating its ability to correctly differentiate between healthy cotton bolls and damaged or pest-infected samples with minimal misclassification.

Performance evaluation metrics including accuracy, precision, recall, and F1-score were used to assess the effectiveness of the classification stage. The CNN-based approach showed improved recall and F1-score compared to traditional classifiers, indicating better sensitivity toward pest patterns and reduced chances of missing infected samples. The YOLO localization model accurately identified the exact region of pest or damage occurrence, enabling region-specific automated spraying rather than full-plant chemical application. The Autoencoder anomaly screening stage further enhanced computational efficiency by forwarding only suspicious samples to deeper models. The overall inference time remained low, making the system suitable for near real-time field monitoring. In addition, the Arduino-based automation module successfully activated neem oil spraying and SMS alert notification only when pest or damage was detected, confirming the effectiveness of the event-driven selective action mechanism.

The experimental observations indicate that the integration of image preprocessing, anomaly detection, object localization, and CNN-based feature learning significantly improves pest detection reliability in cotton crops. Illumination correction and noise removal contributed to more stable predictions in outdoor environments where lighting conditions vary frequently. The CNN effectively captured complex spatial and texture-based disease characteristics, reducing dependency on manual feature engineering methods. Compared with conventional machine learning techniques such as Support Vector Machines and k-Nearest Neighbors, the deep learning pipeline demonstrated superior robustness to scale variation, orientation differences, and environmental noise.

Furthermore, the hybrid combination of deep learning intelligence with IoT-based automation provides a complete end-to-end agricultural monitoring solution rather than a standalone classification tool. The selective activation logic, where spraying and alerts occur only during confirmed pest detection while the system remains idle for healthy crops, reduces chemical wastage, conserves power, and increases farmer trust. However, system performance is influenced by dataset diversity and environmental variability, and limited training samples may reduce generalization to unseen pest types. Future enhancements may include expanding seasonal datasets, integrating environmental sensor data such as humidity and temperature, and deploying lightweight edge-AI models for faster on-field inference. Overall, the proposed Smart Cotton Boll Monitoring System demonstrates strong potential for precision agriculture by enabling early pest

detection, minimizing crop loss, and improving farming efficiency through intelligent and selective automation.



Conclusion:

The Smart Cotton Boll Monitoring and Fertilizer Management System presents a comprehensive, AI-driven, and IoT-enabled solution for precision agriculture. By seamlessly integrating Autoencoder-based anomaly detection,

YOLOv8 object detection, and CNN disease classification, the system is capable of early identification of abnormal cotton bolls, pest infestations, and crop diseases. This early detection allows for timely interventions, including targeted neem oil spraying through an Arduino-controlled automation module, which helps mitigate crop damage before applying fertilizers.

The system's decision and control module intelligently combines AI outputs to determine the optimal course of action, while the fertilizer recommendation module provides condition-based nutrient suggestions tailored to the detected crop health. The notification and alert module, leveraging a Flask backend and Twilio API, ensures farmers receive real-time, actionable updates via SMS and a web dashboard, enabling prompt and informed decision-making.

Through experimental evaluation, the hybrid pipeline demonstrated high accuracy, precision, and recall in detecting both pests and diseases, while the inclusion of the Autoencoder reduced unnecessary YOLO processing, improving overall computational efficiency. The modular architecture of the system makes it scalable, cost-effective, and adaptable to different cotton-growing regions or other crops with minimal retraining.

Overall, this integrated framework represents a robust precision agriculture ecosystem that reduces manual labor, minimizes chemical usage, and enhances crop yield and quality. By combining advanced AI algorithms with IoT automation, the system provides a reliable and sustainable approach to modern farming, paving the way for smarter, data-driven agricultural practices in the future.

Reference:

1.SS et al. (2024) proposed an AI-based agricultural intelligence model to enhance sustainability and precision farming practices. Chen et al. (2025) developed Pest-PVT for multi-class and dense pest detection in field environments. Yadav et al. (2022) evaluated YOLOv5 for detecting volunteer cotton plants across growth stages, confirming its effectiveness for real-time crop monitoring.

2.Sladojevic et al. (2016) demonstrated early success of deep neural networks for leaf-based disease recognition. Mohanty et al. (2016) showed that deep learning achieves high accuracy for image-based plant disease detection. Ferentinos (2018) compared deep learning models and confirmed CNN superiority for disease diagnosis. Lu et al. (2017) applied CNNs for rice disease identification with strong classification performance.

3.Barbedo (2018) analyzed practical factors affecting deep learning performance in plant disease recognition. Kamilaris and Prenafeta-Boldú (2018) presented a survey on deep learning applications in agriculture. Too et al. (2019) demonstrated the benefits of transfer learning for plant disease identification. Gupta et al. (2025) provided a comprehensive review of deep learning-based plant disease detection techniques. Lee et al. (2017) investigated feature extraction mechanisms in deep learning for plant classification.

4.Singh and Gupta (2019) applied machine learning algorithms for plant disease classification. Amara et al. (2017), Brahimi et al. (2017), Zhang et al. (2019), Durmus et al. (2020), Islam et al. (2020), and Khan et al. (2020) implemented CNN-

based approaches for disease detection in various crops, demonstrating consistent improvements in classification accuracy. Sannakki et al. (2013) introduced neural network-based grape disease classification as an early contribution in this field.

6.Gowtham and Jebakumar (2022) integrated IoT with machine learning for automated spraying systems. Patil and Kumar (2023) reviewed advancements in combining deep learning with IoT for automated plant disease detection

7.M. J. Islam, A. S. M. S. Rahman, M. N. Islam, and M. M. Hasan, "Leaf disease detection using convolutional neural networks," International Journal of Engineering Research & Technology, vol. 9, no. 10, pp. 234–240, Oct. 2020.

8. N. D. Khan, M. Shah, and A. Ali, "Plant disease detection using image processing and machine learning," International Journal of Advanced Computer Science and Applications, vol. 11, no. 6, pp. 492–498, June 2020.

9.H. Durmus, E. O. Gunes, and M. Kirci, "Disease detection on the leaves of the tomato plants by using deep learning," AgriEngineering, vol. 2, no. 3, pp. 316–331, Sept. 2020.

10 .S. Zhang, C. Zhang, X. Wang, and Y. Shi, "Cucumber leaf disease identification with deep convolutional neural networks," Applied Sciences, vol. 9, no. 17, pp. 1–14, Sept. 2019.

11. R. Sannakki, V. S. Rajpurohit, V. B. Nargund, and P. Kulkarni, "Diagnosis and classification of grape leaf diseases using neural networks," International Journal of Computer Applications, vol. 975, no. 6, pp. 1–5, 2013.

12. R. P. Patil and S. Kumar, "Advances in automated plant disease detection using deep learning and IoT," IEEE Access, vol. 11, pp. 45678–45692, 2023.