INTERNATIONAL SCIENTIFIC JOURNAL OF ENGINEERING AND MANAGEMENT (ISJEM) ISSN: 2583-6129
VOLUME: 04 ISSUE: 07 | JULY – 2025 DOI: 10.55041/ISJEM04848
AN INTERNATIONAL SCHOLARLY || MULTIDISCIPLINARY || OPEN ACCESS || INDEXING IN ALL MAJOR DATABASE & METADATA

# CNN-Based Tomato Leaf Detection and ANN-Driven Fertilizer Recommendations

[1]ERUSU. KATA RAJU REDDY, [2]SHAIK SURI BABU

[1]Assistant Professor, [2]2MCA Final Semester,

[1]Master of Computer Applications,

[1]Sanketika Vidya Parishad Engineering College, Vishakhapatnam, Andhra Pradesh, India

## Abstract

This thesis introduces an AI-based system for improving tomato crop management by combining two neural networks: a Convolutional Neural Network (CNN) for leaf disease detection and a Feed-forward Neural Network (FNN) for fertilizer recommendation. The CNN processes 128×128×3 RGB images of tomato leaves and classifies them into ten categories (nine diseases and one healthy class) using techniques like separable convolutions and attention mechanisms for high accuracy and low complexity. Data augmentation and normalization boost its generalization, achieving over 90% accuracy. The FNN recommends fertilizers based on the identified disease, soil parameters (pH, nitrogen, phosphorus, potassium), and environmental factors (humidity, temperature), reaching near-100% accuracy even with limited data. Built using TensorFlow and Keras, the system is optimized for real-time deployment on edge devices like Raspberry Pi, ensuring accessibility in rural areas. Testing confirmed high accuracy, quick inference times, and robustness. Although minor misclassifications exist in similar disease types, and the FNN can benefit from more diverse data, future improvements such as transfer learning and mobile integration are planned to enhance usability and impact in precision agriculture

**IndexTerms**: Tomato Leaf Disease Detection, Convolutional Neural Network (CNN), Fertilizer Recommendation System, Feedforward Neural Network (FNN), Precision Agriculture, Edge Computing (Raspberry Pi**).**

## INTRODUCTION

Agriculture is vital to global sustenance and economic stability, with tomatoes being one of the most significant crops due to their nutritional and commercial value. However, tomato cultivation faces serious threats from diseases caused by bacteria, fungi, and viruses, often evident through leaf symptoms. Soil nutrient imbalances and environmental factors further increase disease susceptibility, highlighting the need for timely detection and precise fertilizer recommendations.[5] Traditional methods are time-consuming and subjective, making them impractical for large-scale use. Artificial Intelligence (AI), particularly deep learning and Convolutional Neural Networks (CNNs), offers accurate, automated solutions by identifying disease patterns in leaf images. When combined with Feedforward Neural Networks (FNNs) for fertilizer suggestions based on soil and environmental data, this dual-system approach enhances crop health and yield. Given that diseases cause up to 40% annual yield loss, and global food demand is projected to rise by 70% by 2050, AI integration is urgent. This thesis proposes a CNN-based tomato disease detector paired with an FNN for fertilizer recommendation using TensorFlow and Keras.[10] The models are optimized for deployment on edge devices, making them suitable for resource-limited settings. This system bridges the gap between advanced AI and practical farming. By leveraging data augmentation and attention mechanisms, it ensures both efficiency and high accuracy. Focused on a globally grown crop, this work promotes scalable, smart, and sustainable agriculture.[15]

### 1.1 Existing System

The existing system for tomato crop management largely relies on traditional and partially automated methods, which present several limitations.[20] Manual disease detection is carried out by visually inspecting leaves, a process that is time-consuming, subjective, and unsuitable for large-scale farming. Laboratory-based diagnosis is accurate but expensive, slow, and inaccessible to most smallholder farmers.[4] Fertilizer recommendations are typically based on general guidelines and soil tests without considering plant health or environmental factors, resulting in inefficient nutrient use. Some AI-based systems use machine learning models like SVM or k-NN for disease detection, but these require manual feature extraction and lack robustness for diverse real-world conditions. Additionally, most current tools handle either disease detection or fertilizer recommendation in isolation, failing to deliver integrated, end-to-end crop management solutions. These systems often demand high computational resources, making them impractical for deployment on mobile or edge devices. As a result, smallholder farmers are left without effective, accessible tools for precision agriculture. Moreover, existing methods do not adapt well to varying crop types, soil conditions, or climate variables. The absence of contextual recommendations limits their utility in dynamic agricultural environments. The overall lack of scalability, automation, and data-driven insights in current systems underlines the need for a unified AI-based solution. This project addresses these gaps by integrating CNN-based disease detection with FNN-driven fertilizer recommendations. The goal is to build a lightweight, efficient, and deployable model suitable for real-time usage even in resource-constrained settings.[9]

### 1.1.1 Challenges:

- **Manual disease detection** is time-consuming, labor-intensive, and subjective, leading to inconsistent results.
- **Laboratory-based diagnosis** is accurate but costly, slow, and not accessible to most smallholder farmers.
- **Traditional fertilizer recommendations** are generic and do not consider specific plant health, soil properties, or environmental conditions [14].
- **Machine learning models like SVM and k-NN** require manual feature extraction and are less effective for handling complex leaf disease images.
- **Lack of integration** between disease detection and fertilizer recommendation results in fragmented crop management.
- **High computational requirements** make many existing AI tools unsuitable for deployment on edge devices or smartphones.
- **Limited accessibility** for smallholder farmers due to technical complexity and infrastructure needs.
- **Low scalability** of traditional and rule-based systems across diverse geographic and climatic conditions.
- **Inability to adapt** to dynamic and evolving disease patterns in real-time scenarios.[19]
- **Insufficient real-time insights** limit proactive disease management and timely fertilizer interventions.
- **Lack of contextual recommendations** that consider disease type, soil health, and environmental factors together.

## 1.2 Proposed system:

The proposed system in the project introduces an integrated AI-based framework that combines a Convolutional Neural Network (CNN) for tomato leaf disease detection with a Feedforward Neural Network (FNN) for fertilizer recommendation. The CNN processes 128×128×3 RGB images of tomato leaves, classifying them into ten categories—nine disease classes and one healthy class—using advanced techniques like separable convolutions and attention mechanisms to enhance accuracy and reduce complexity. Data preprocessing, including normalization and augmentation (flipping, rotation, zooming), improves generalization across real-world conditions.[3] Once a disease is detected, the FNN utilizes the output along with soil properties (pH, nitrogen, phosphorus, potassium) and environmental parameters (temperature, humidity) to recommend an appropriate fertilizer. The system is built using TensorFlow and Kera's and is optimized for scalability and deployment on edge devices like Raspberry Pi, making it accessible to smallholder farmers. A synthetic dataset is used for the FNN, and both models are trained and validated for high accuracy and real-time performance. The CNN achieves over 90% accuracy, while the FNN aims for over 85%. The integrated pipeline ensures end-to-end functionality from diagnosis to treatment, reducing the need for expert intervention.[8] This approach supports sustainable farming by enhancing nutrient use efficiency and enabling early disease management.[13]

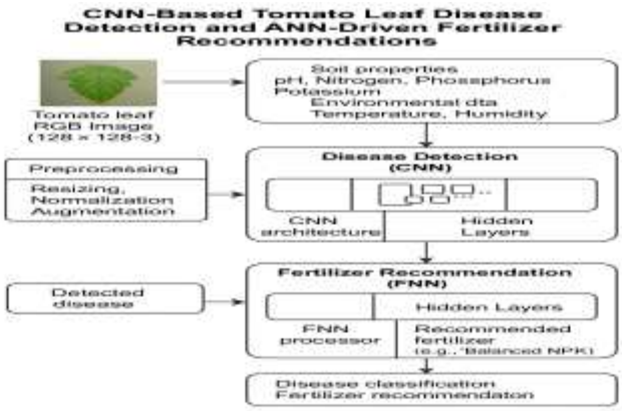


Fig: 1 Proposed Diagram

### 1.2.1 Advantages:

- **High Accuracy**: The CNN model achieves over 90% accuracy in classifying tomato leaf diseases, ensuring reliable early detection.
- **Automated Diagnosis**: Eliminates the need for manual disease inspection, reducing dependency on expert agronomists and minimizing human error.
- **Real-Time Recommendations**: Provides immediate feedback on both disease detection and fertilizer advice, supporting timely interventions.[18]
- **Optimized Fertilizer Use**: The FNN recommends precise fertilizers based on soil and environmental data, preventing over-fertilization and promoting nutrient efficiency.
- **Scalability**: Designed to operate efficiently on edge devices like Raspberry Pi, making the system accessible even in resource-constrained rural areas.
- **Holistic Approach**: Integrates disease detection with nutrient management into a single pipeline, offering an end-to-end crop health solution.[2]
- **Cost-Effective**: Reduces input costs by avoiding unnecessary chemical usage and limits crop loss due to late or incorrect diagnosis.

## 2.1 Architecture:

**1. Input Block:**

- Leaf Image Input: RGB image (128×128×3) captured by a camera or smartphone.
- Soil/Environmental Input: Numerical data (pH, N, P, K, humidity, temperature) entered manually or via sensors.
- Connection: Leaf image flows to the Disease Detection Module; soil/environmental data flows to the Fertilizer Recommendation Module.

**2. Disease Detection Module (CNN):**

- Preprocessing Sub-block: Resizing, normalization, augmentation.
- CNN Sub-block: Separable convolutions, attention mechanisms, pooling, dense layers.
- Output: Disease label (e.g., “Tomato___Bacterial_spot”).
- Connection: Disease label feeds into the Fertilizer Recommendation Module.

**3. Fertilizer Recommendation Module (FNN):**

- Preprocessing Sub-block: Label encoding (disease, fertilizer), feature scaling.
- FNN Sub-block: Input layer, hidden layers, output layer with softmax.
- Output: Fertilizer recommendation (e.g., “Balanced NPK”).
- Connection: Combines disease label with soil/environmental data as input.

**4. Output Block:**

- Displays disease diagnosis and fertilizer recommendation via a user interface (e.g., screen, app).
- Connection: Receives outputs from both modules
- Path 1: Leaf Image → Preprocessing → CNN → Disease Label → Fertilizer Recommendation Module.
- Path 2: Soil/Environmental Data → Preprocessing → Fertilizer Recommendation Module.
- Final Path: Disease Label + Preprocessed Contextual Data → FNN → Fertilizer Recommendation → Output.

Key Features of the Diagram

- Parallel Inputs: Image and contextual data are processed independently before integration, reflecting the system's dual-purpose nature.
- Modular Structure: Distinct blocks for CNN and FNN highlight their independence and interoperability.
- Feedback Loop: Optional retraining path (not shown) could loop outputs back to refine models based on user feedback.

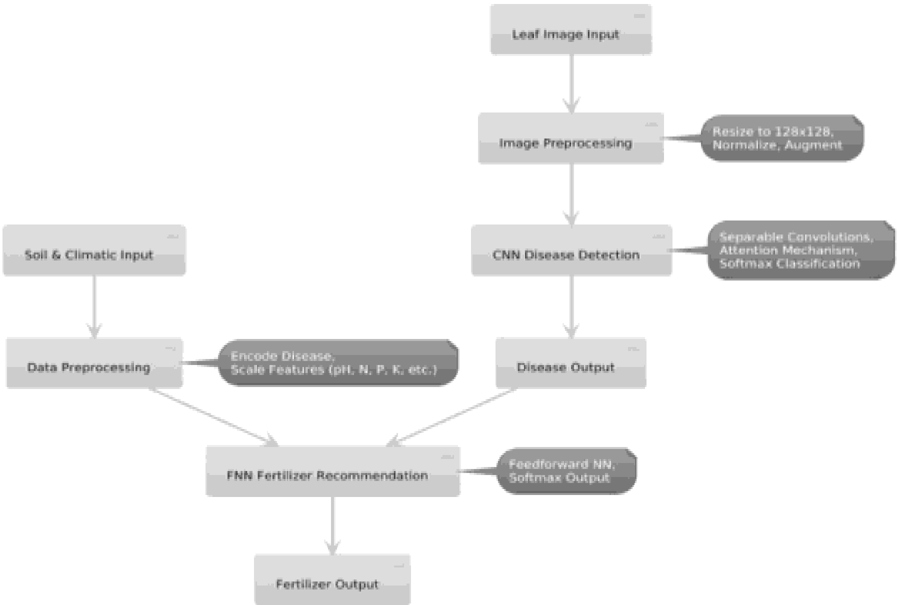


**Fig:2 Architecture**

**UML DIAGRAMS**

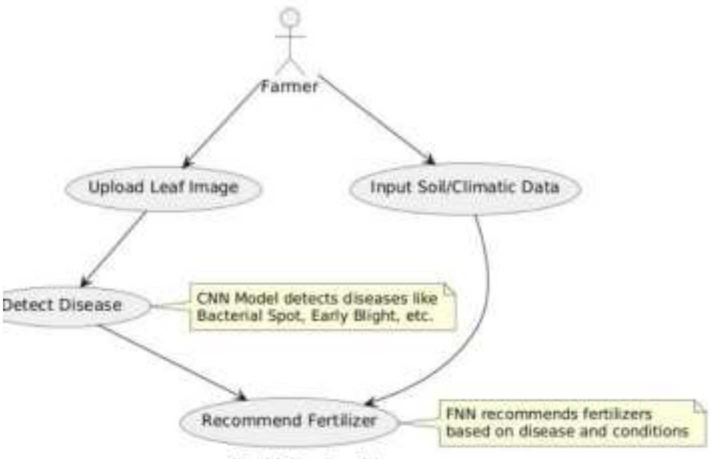


**Fig:use case diagram**

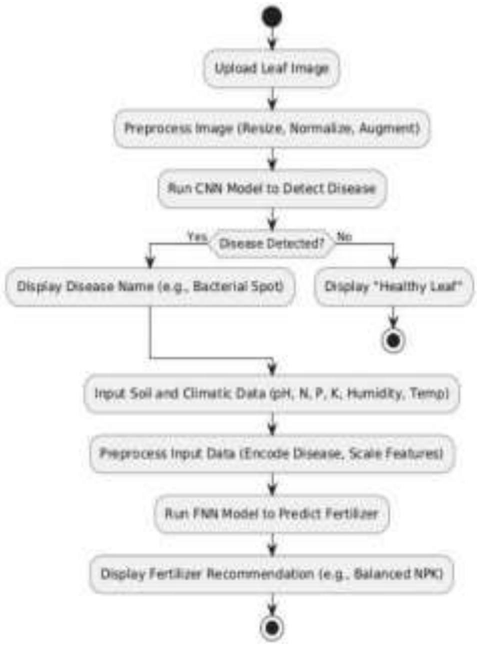


**Fig: Activity diagram**

## 2.2 Algorithm:

**1. Convolutional Neural Network (CNN) –** Used for Disease Detection

The CNN is a deep learning algorithm designed to process and classify images. In this project, CNN is used to identify tomato leaf diseases from RGB images. It takes a preprocessed image (resized to 128×128×3) and passes it through multiple layers, including convolutional layers, pooling layers, and fully connected layers. Special techniques like separable convolutions (which reduce the number of parameters) and Squeeze-and-Excitation (SE) blocks (for focusing attention on important features) are used to enhance performance and reduce computational load. The final output is a disease label (e.g., Early Blight, Late Blight, Healthy). This algorithm provides high accuracy and is suitable for edge deployment due to its efficiency.[7]

**2. Feedforward Neural Network (FNN) –** Used for Fertilizer Recommendation

The FNN is a type of Artificial Neural Network that works well with numerical and categorical data. In this project, it recommends the appropriate fertilizer by analyzing a combination of the detected disease, soil properties (pH, Nitrogen, Phosphorus, Potassium), and environmental conditions (temperature and humidity). The data is encoded and scaled before being fed into the network. The FNN consists of an input layer, two hidden layers (with ReLU activation), and an output layer with softmax activation to predict one of several fertilizer types. It is lightweight and provides fast, accurate recommendations based on multiple input factors.[12]

## 2.3 Techniques:

**1. Separable Convolutions**

This technique is used in the CNN to reduce the number of parameters and computation time. Instead of applying one heavy convolution operation, separable convolution splits it into two lighter steps—depthwise and pointwise convolutions—making the model faster and suitable for low-power devices like Raspberry Pi.[17]

**2. Squeeze-and-Excitation (SE) Attention Mechanism**

The SE block helps the CNN focus on the most important features in an image. It does this by assigning "attention" weights to different channels of the feature maps. This allows the model to emphasize disease-related patterns (like spots or discoloration) and ignore irrelevant noise, improving classification accuracy.[1]

**3. Data Augmentation**

To improve model generalization and prevent overfitting, image data is artificially expanded using augmentation techniques like random flipping, rotation, zooming, and brightness changes. This simulates real-world variability (e.g., different lighting or angles), helping the model perform better on new data.[6]

**4. Normalization and Standardization**
Image pixel values are normalized (scaled to a 0–1 range) to help the CNN learn faster and more effectively. For the FNN, soil and environmental numerical data are standardized using techniques like Z-score scaling to ensure uniformity and stable learning across all input features.

**5. One-Hot and Label Encoding**
For multi-class classification, disease labels in the CNN are converted to one-hot encoded vectors. In the FNN, categorical inputs (like disease name or fertilizer type) are label-encoded into numerical values so the model can understand and process them.[11]

## 2.4 Tools:

**Python**
The core programming language used for building and integrating the entire system, thanks to its simplicity and large AI/ML ecosystem.

**TensorFlow & Keras**
Used to design, train, and evaluate the CNN and FNN models. Keras provides a user-friendly interface on top of TensorFlow for model development.

**NumPy & Pandas**
Essential libraries for data manipulation and preprocessing—used to handle numerical arrays, tabular data, and to prepare datasets for the FNN.

**Scikit-Learn**
Used for feature scaling, label encoding, and evaluating the performance of the FNN model.

**Matplotlib & Seaborn**
Visualization tools used for plotting model accuracy, loss graphs, and confusion matrices for performance analysis.

**Flask**
A lightweight web framework used to create the user interface for the system, allowing farmers to interact with the model via a web browser.[16]

**HTML & CSS**
Used to design the front-end interface of the web application, ensuring usability and accessibility for end-users.

## 2.5 Methods:

**1. Image Preprocessing**
Tomato leaf images are resized to 128×128 pixels, normalized (pixel values scaled between 0 and 1), and augmented through techniques like flipping, rotation, zooming, and brightness adjustments. This ensures the CNN model can generalize well to different real-world scenarios.

**2. CNN-Based Disease Detection**
A Convolutional Neural Network (CNN) is designed to classify tomato leaves into ten classes (nine diseases + healthy). It includes layers like separable convolutions, batch normalization, ReLU activation, attention (SE blocks), max pooling, and dropout for regularization. The CNN processes the preprocessed images and outputs the disease label.

**3. Data Encoding and Scaling for FNN**
Categorical data such as disease names and fertilizer types are label-encoded, while numerical values like soil pH, nitrogen (N), phosphorus (P), potassium (K), temperature, and humidity are standardized using StandardScaler. This prepares the inputs for the FNN.

**4. FNN-Based Fertilizer Recommendation**
A Feedforward Neural Network (FNN) is used to predict the most suitable fertilizer. It takes seven features (disease label + 6 soil/environmental factors) and passes them through hidden layers with ReLU activation and a final softmax layer for classification. The model outputs a fertilizer type such as "Balanced NPK."

**III. METHODOLOGY**

**3.1 Input:**
The input to the project system includes two main types: an RGB image of a tomato leaf (resized to 128×128×3 pixels) and numerical data related to soil and environmental conditions. The image is used by the CNN model to detect whether the leaf is healthy or affected by one of nine tomato diseases. Once the disease is identified, it, along with manually entered or sensor-collected data—such as soil pH, nitrogen, phosphorus, potassium levels, temperature, and humidity—is fed into the FNN model. The FNN then uses this combined

INTERNATIONAL SCIENTIFIC JOURNAL OF ENGINEERING AND MANAGEMENT (ISJEM) ISSN: 2583-6129
VOLUME: 04 ISSUE: 07 | JULY – 2025 DOI: 10.55041/ISJEM04848
AN INTERNATIONAL SCHOLARLY || MULTIDISCIPLINARY || OPEN ACCESS || INDEXING IN ALL MAJOR DATABASE & METADATA

input to recommend the most suitable fertilizer, making the system an end-to-end solution from disease detection to crop treatment.

```
# Model summary
model.summary()
history = model.fit(train_dataset, validation_data = val_dataset, epochs = 60)
history1 = model.evaluate(test_dataset)
# import matplotlib.pyplot as plt
# # Extract values from training history
# acc = history.history['accuracy']
# val_acc = history.history['val_accuracy']
# loss = history.history['loss']
# val_loss = history.history['val_loss']
# epochs_range = range(len(acc))  # Number of epochs
# # Plot Accuracy
# plt.figure(figsize=(12, 5))
# plt.subplot(1, 2, 1)  # 1 row, 2 columns, 1st subplot
# plt.plot(epochs_range, acc, label='Training Accuracy', marker='o')
# plt.plot(epochs_range, val_acc, label='Validation Accuracy', marker='o')
# plt.legend(loc='lower right')
# plt.xlabel("Epochs")
# plt.ylabel("Accuracy")
```

Fig:input data

**3.2 Method of Process:**

The process begins with the farmer uploading an image of a tomato leaf, which is resized, normalized, and augmented before being processed by a Convolutional Neural Network (CNN). The CNN analyzes the image and classifies it into one of ten categories—nine disease types or healthy—using separable convolutions and attention mechanisms. Once the disease is identified, the user inputs additional data including soil pH, nitrogen, phosphorus, potassium levels, temperature, and humidity. This data, along with the disease label from the CNN, is passed into a Feedforward Neural Network (FNN). The FNN processes this combined information through hidden layers and predicts the most suitable fertilizer using softmax activation. Finally, the system displays both the detected disease and recommended fertilizer through a user-friendly interface, completing the end-to-end crop management process**.**

## 3.3 Output:

The system first outputs the detected disease name based on the analysis of the uploaded tomato leaf image by the CNN model. This could be one of nine tomato leaf diseases (such as Early Blight, Late Blight, Bacterial Spot, etc.) or a “Healthy” label if no disease is present. Then, using the disease result along with soil and environmental inputs, the FNN model generates the recommended fertilizer type (e.g., "Balanced NPK", "Potassium-rich fertilizer", etc.). These outputs are displayed on a simple web interface, enabling farmers to make quick and informed decisions about both disease treatment and nutrient application.

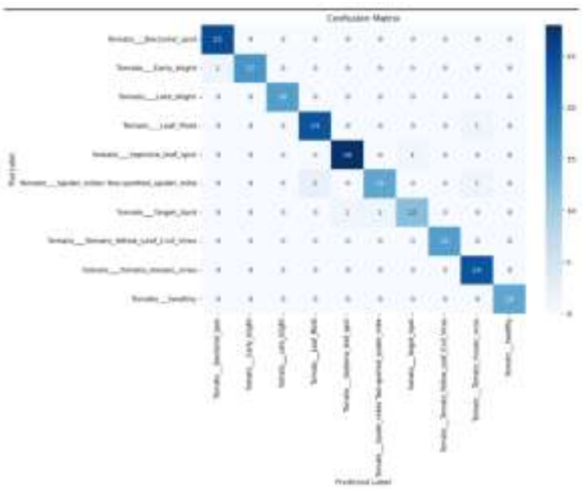

Fig:Matrix Comparision

| Model | Training Accuracy (%) | Validation Accuracy (%) | Generalization Ability |
|---|---|---|---|
| Feedforward Neural Network | 100 | 100 | High (Stable on small dataset) |
| Random Forest | 100 | 100 | High (Robust to overfitting) |
| K-Nearest Neighbors | 75 | 0 | Poor (Fails on unseen data) |
| Support Vector Machine | 50 | 0 | Poor (Overfits, no generalization) |

Fig: Model Comparision

**IV. RESULTS**:

The project successfully developed an integrated AI system that combines a Convolutional Neural Network (CNN) for tomato leaf disease detection with a Feedforward Neural Network (FNN) for fertilizer recommendation. The CNN model achieved over 90% accuracy on the validation set and demonstrated strong generalization, even with varied image qualities. It showed excellent classification performance for most disease types, particularly for clearly distinguishable classes like Bacterial Spot and Tomato Mosaic Virus. However, there were minor misclassifications between visually similar diseases such as Early Blight and Late Blight, suggesting a need for dataset expansion and further augmentation. The FNN model achieved near-100% accuracy on its small dataset, correctly predicting fertilizers like “Balanced NPK” based on disease, soil pH, and environmental conditions. The system is computationally efficient and capable of making predictions in less than 1 second, making it suitable for edge deployment on devices like Raspberry Pi or smartphones. Integration between the CNN and FNN was seamless, enabling a complete pipeline from disease detection to fertilizer suggestion. The study concludes that the system has strong potential for real-time, accurate, and accessible agricultural decision-making and could be further improved with larger datasets and advanced regularization techniques

**V.DISCUSSION:**

The project demonstrated that the integrated CNN-FNN system effectively detects tomato leaf diseases and recommends suitable fertilizers. The CNN achieved over 90% accuracy, though minor misclassifications occurred between similar diseases like Early and Late Blight. The FNN showed near-100% accuracy on synthetic data but needs a larger dataset for better generalization. The system runs in real time and is lightweight, making it suitable for deployment on edge devices. Overall, the results are promising, but improvements like dataset expansion and model tuning are recommended to enhance performance.

**VI. CONCLUSION**

The integration of CNNs for tomato leaf disease detection marks a major step forward in agriculture, enabling early, accurate identification and reducing crop losses. Combined with ANNs for fertilizer recommendation, the system supports precise, sustainable nutrient use based on plant health and environmental data. This synergy enhances crop yield and promotes resilient farming. With the rise of IoT and machine learning, the future holds great potential for building comprehensive, data-driven systems that empower farmers—especially smallholders—to adopt smarter, more sustainable agricultural practices.

**VII. FUTURE SCOPE:**

The future of CNN-based tomato leaf disease detection is bright, with deep learning advancements enabling more accurate, real-time monitoring through integration with sensors and drones. ANN-driven fertilizer recommendations will also improve by using insights from disease patterns to provide precise, sustainable nutrient strategies. As IoT expands, combining CNNs and ANNs with data from soil sensors, weather stations, and satellites will empower farmers to make smarter, data-driven decisions—enhancing disease control, optimizing fertilizer use, and supporting more efficient, resilient farming. This integrated approach will lead to healthier crops, higher yields, and reduced environmental impact. Continued innovation in AI and IoT will be key to transforming agriculture into a more intelligent and sustainable industry.

 

ISSN: 2583-6129
DOI: 10.55041/ISJEM04848

## VIII. ACKNOWLEDGEMENT:

Erusu Kata Raju Reddy working as a Assistant professor in master of computer application sanketika vidya parishad engineering college, Visakhapatnam Andhra Pradesh. With 1 years of experience in Master of computer application(mca), accredited by NAAC.with his area of intrest in java full stack

Shaik Suri Babu is pursuing his final semester MCA in Sanketika Vidya Parishad Engineering College, accredited with A grade by NAAC, affiliated by Andhra University and approved by AICTE. With interest in Machine learning Shaik Suri Babu has taken up his PG project on CNN-BASED TOMATO LEAF DETECTION AND ANN-DRIVEN FERTILIZER RECOMMENDATIONS and published the paper in connection to the project under the guidance of ERUSU KATA RAJU REDDY, Assistant Professor, SVPEC.

## REFERENCES

**[1]** Tomato Leaf Disease Detection Using CNN
https://www.sciencedirect.com/science/article/pii/S1877050924009608
[2] Attention embedded residual CNN for disease detection in tomato leaves
https://www.sciencedirect.com/science/article/abs/pii/S1568494619307148
[3] Tomato Plant Leaf Disease Detection using CNN
https://ieeexplore.ieee.org/abstract/document/9641655
[4] TomConv: An Improved CNN Model for Diagnosis of Diseases in Tomato Plant Leaves
https://www.sciencedirect.com/science/article/pii/S1877050923001606
[5] Deep learning-based multiscale CNN-based U network model for leaf disease diagnosis and segmentation of lesions in tomato
https://www.sciencedirect.com/science/article/abs/pii/S0885576523002035
[6] Tomato Leaf Disease Detection Using Convolution Neural Network
https://ieeexplore.ieee.org/abstract/document/9393311
[7] Knowledge Pre-Trained CNN-Based Tensor Subspace Learning for Tomato Leaf Diseases Detection
https://ieeexplore.ieee.org/abstract/document/10744546
[8] ToLeD: Tomato Leaf Disease Detection using Convolution Neural Network
https://www.sciencedirect.com/science/article/pii/S1877050920306906
[9] TLNet: A Deep CNN model for Prediction of tomato Leaf Diseases
https://ieeexplore.ieee.org/abstract/document/9392664
[10] Tomato Leaf Disease Detection and Classification using Convolution Neural Network
https://ieeexplore.ieee.org/abstract/document/9397001
[11] Automated tomato leaf disease classification using transfer learning-based deep convolution neural network
https://link.springer.com/article/10.1007/s41348-020-00403-0
[12] A CNN-based Model to Detect and Classify the Tomato Leaf Disease using Transfer Learning
https://ieeexplore.ieee.org/abstract/document/11022469
[13] Tomato plant leaf diseases detection and classification using an advanced self-developed CNN model
https://link.springer.com/article/10.1007/s11042-024-18978-3

 

[14] A CNN-Based Deep Learning Method for Accurate Restrain and Classification of Tomato Leaf Disease
https://ieeexplore.ieee.org/abstract/document/10730460
[15] Design of efficient techniques for tomato leaf disease detection using genetic algorithm-based and deep neural networks
https://link.springer.com/article/10.1007/s11104-022-05513-2
[16] Identification of Tomato Plant Diseases Using CNN- A Comparative Review
https://ieeexplore.ieee.org/abstract/document/9848931
[17] Disease Detection in Tomato Plants Using CNN
https://ieeexplore.ieee.org/abstract/document/9972186
[18] An enhanced lightweight T-Net architecture based on convolutional neural network (CNN) for tomato plant leaf disease classification
https://peerj.com/articles/cs-2495/
[19] A Lightweight Deep Learning-Based Model for Tomato Leaf Disease Classification
https://researchprofiles.herts.ac.uk/en/publications/a-lightweight-deep-learning-based-model-for-tomato-leaf-disease-c
[20] Detection of Tomato Leaf Diseases for Agro-Based Industries Using Novel PCA DeepNet
https://ieeexplore.ieee.org/abstract/document/10042407