International Journal of Leading Research Publication (IJLRP)

E-ISSN: 2582-8010 ● Website: www.ijlrp.com ● Email: editor@ijlrp.com

ADAPTIVE BOUNDARY LEARNING AND ETENSN-AWARE MOBILE TOMATO CROP DISEASE MONITORING FOR PRECISION FARMING

Lalitha Reddy, Badam

Software Developer (Full-Time) & Adjunct Professor (Part-Time)
Computer Science Department
Southern New Hampshire University, USA
lalithabadambvsr@gmail.com

Abstract:
Tomato cultivation faces major yield losses due to fungal, bacterial, and insect diseases. However, existing methods lacked adaptive feature similarity boundary learning, reducing classification accuracy. Hence, this research work presents an adaptive similarity boundary learning and ETENSN-aware mobile tomato crop disease monitoring system for precision farming. Initially, input leaf images are pre-processed based on WF, and data are balanced. Then, the complex backgrounds are removed using the GMM approach. After the removal of the background, the RGB image is converted to HSV. Meanwhile, from the background-removed outcome, the vein structures are extracted using HGSSMM. From color converted image and the extracted vein structure, the features are extracted. Next, the feature similarity boundary is extracted from the extracted features using the ABDNN approach. After that, the extracted features, the pre-processed image, and the extracted similarity boundary are given as input to the ETENSN for disease classification. At last, for the obtained diseases, the nutrient is recommended using FTMIS to further preserve the farming. Experimental evaluation shows that the model attains 99.6235% accuracy, which is superior to existing methods.

Keywords: Agriculture technology, Crop disease detection, Image processing, Mobile application, Tomato Leaf, Approximate Bhattacharyya Distanced Nearest Neighbor (ABDNN), and Efficient Transferred Elastic Net and Softplus Network (ETENSN).

1. INTRODUCTION

Agriculture plays a crucial role in global livelihoods by supporting food production, employment, and economic development (Ruhad et al., 2025). Among agricultural crops, tomato is widely cultivated and consumed due to its nutritional and commercial importance (Hossain et al., 2023), and improving tomato productivity significantly contributes to economic growth and poverty reduction (Al-Shamasneh & Ibrahim, 2024). However, tomato plants are highly susceptible to bacterial, viral, and fungal diseases, which reduce crop yield and quality. Early disease identification is therefore essential to minimize production loss (Nguyen et al., 2023)(Zhang et al., 2024). Traditional manual disease detection is labor-intensive and often leads to incorrect pesticide usage by inexperienced farmers, further worsening disease spread (Yu et al., 2024)(Deng et al., 2023). Recently, Artificial Intelligence (AI), particularly Machine Learning and Deep Learning techniques, has been applied for automated crop disease prediction and monitoring (Oni & Prama, 2025)(Chen et al., 2025). Although existing models, such as Convolutional Neural Network (CNN), Convolutional Block Attention Module (CBAM), and Support Vector Machine (SVM), enabled mobile-based disease detection, they still struggled to capture fine-grained disease variations (Ghosh et al., 2025). Moreover, none of the prevailing works focused on learning adaptive feature similarity boundaries among disease classes of subtle visual changes. Hence, this research work develops an ETENSN approach-based tomato crop disease monitoring system.

1.1 Research overview

Research gap: None of the existing research works learned adaptive feature similarity boundaries among disease classes of subtle visual changes, resulting in reduced classification accuracy.

Problem statement: The limitations of the prevailing works are discussed as follows,

- Existing (Nawaz et al., 2022) did not provide nutrient recommendations or decision-support mechanisms, thereby limiting its practical usefulness for tomato crop improvement.
- Disease classification without adequately addressing background effects in (Harakannanavar et al., 2022) led to reduced classification performance and reliability.
- Most existing studies did not extract vein-related information.
- Most of the prevailing works did not support a variety of diseases occurred in the tomato crop, which might create a poor generalizability issue.

Significance and scope of the study: The study improves tomato disease detection using adaptive similarity boundary learning and ETENSN-based mobile monitoring. It supports real-time disease identification and nutrient recommendation for precision farming.

Research aim and objectives: The objectives of the proposed research are listed as,

- Concerning the ABDNN approach, an adaptive feature similarity boundary is extracted to improve the decision accuracy.
- The nutrients are recommended for the diseased crop by using the Fuzzy Tilted Maxout Inference System (FTMIS) approach to further preserve the crop.
- The Gaussian Mixture Model (GMM) is used to remove the complex backgrounds.
- To extract the vein structure by using the Hessian Gaussian Scale-Space Matrix Method (HGSSMM).
- To provide better generalizability by using the ETENSN approach.

The structure of the paper is organized as follows: In section 2, the existing works are discussed, in section 3, the proposed methodology is given, in section 4, the results of the research are discussed, and in section 5, the paper is concluded with future scope.

2. LITERATURE REVIEW

(Nawaz et al., 2022) represented a robust Deep Learning (DL)-based model for tomato plant leaf disease classification. Initially, the annotations of the suspected images were generated using Region of Interest (RoI). At last, the Faster Recurrent Convolutional Neural Network (Fast-RCNN) was used for predicting the tomato plant leaf anomalies. Experimental evaluation showed that the model attained higher performance than the prevailing methods.

(Harakannanavar et al., 2022) developed a plant leaf disease detection system using computer vision and machine learning algorithms. Initially, the tomato leaves were resized, and Histogram Equalization (HE) was used. Then, the K-Means (KM) approach was used. Next, the features were extracted from the grouped data, and the disease was classified using ML approaches. The presented approach attained higher performance.

(Alzahrani & Alsaade, 2023) suggested an early detection and recognition framework for tomato leaf disease prediction using a DL approach. At first, the input data was pre-processed. Then, the DL model was used for the early detection of disease. Experimental findings of the study showed that the model attained higher classification performance than the existing models. Nonetheless, the model failed to capture the fine-grained details of the image.

(Khasawneh et al., 2022) recommended an automatic detection of tomato diseases using a Deep Transfer Learning (DTL) approach. The input image was directly given as input to the Convolutional Neural Network (CNN) model. The experimental result showed that the model was highly feasible for smartphone based application. Improper handling of the mobile data that consisted of noise might degrade the performance of the model.

(Tarek et al., 2022) introduced an optimized deep learning algorithm for tomato leaf disease detection with hardware deployment. The input data was given to the DL model for the classification of tomato leaf disease. The evaluation of the study showed that the presented model achieved higher performance than the existing research works. The model only classified the less number of diseases, which might produce a poor generalizability.

3. PROPOSED ADAPTIVE FEATURE SIMILARITY BOUNDARY EXTRACTION-AWARE TOMATO CROP DISEASE CLASSIFICATION SYSTEM FOR MOBILE APPLICATION

This paper presents an enhanced crop disease classification system for a mobile application. The block diagram of the proposed research framework is shown in Figure 1.

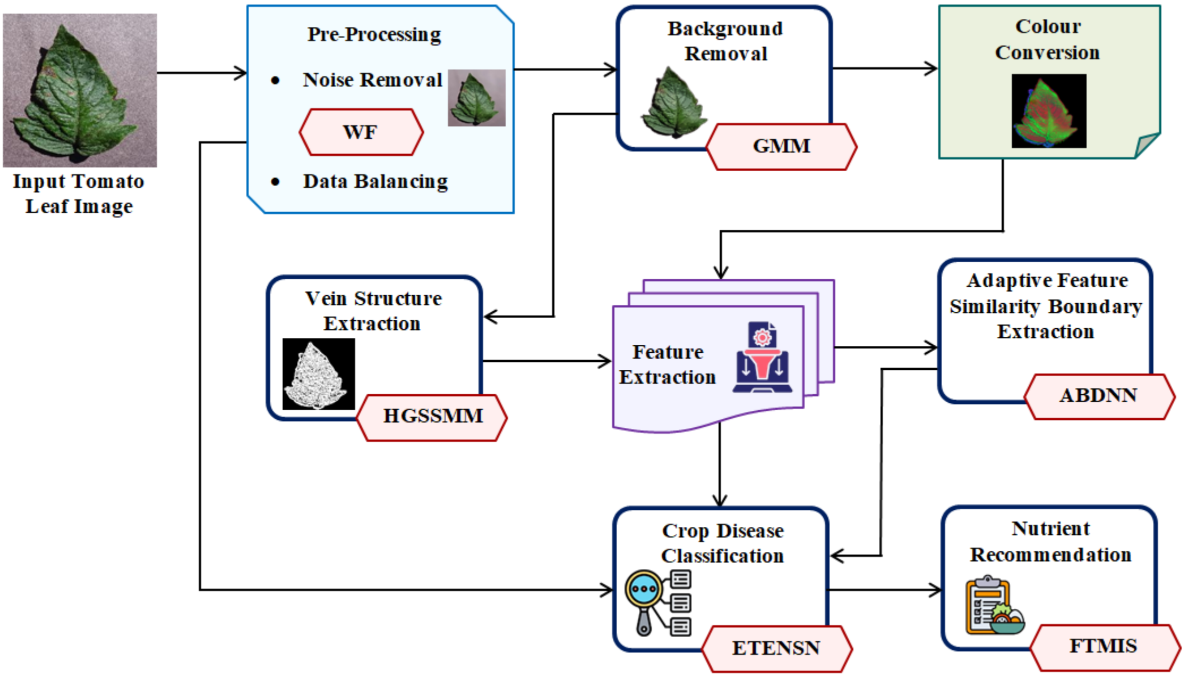


Figure 1: Block diagram for the proposed research framework

3.1 Input tomato leaf image

Initially, the tomato leaf images are collected from public sources.

$$F_d = \{f_1, f_2, f_3, \ldots\ldots, f_z\}$$

(1)

Where, F_d indicates the tomato leaf image dataset, and z -number of leaf images are denoted as f_z.

3.2 Pre-processing

Here, the quality of F_d is increased by removing noises in F_d using the Wiener Filter (WF). The conventional WF adjusts itself based on local image statistics. Also, for varying noise levels, the WF effectively removes the noise.

$$W_f = \frac{f_i^*(x, y)}{|f_i(x, y)|^2 + \Gamma}$$

(2) Where, W_f specifies the noise-removed image, $f_i(x, y)$ is the degradation function of F_d, x and y are the pixels, $f_i^*(x, y)$ specifies the complex conjugate of f_i, and Γ refers to

the noise-to-signal ratio constant. Then, the set of W_f is balanced to reduce the overfitting risk based on flipping, rotation, scaling, translation, shearing, and cropping operations. At last, the pre-processed data is declared as (W_{pr}).

3.3 Background removal

In this section, the complex backgrounds of (W_{pr}) is removed using the Gaussian Mixture Model (GMM). The conventional GMM approach effectively separates the leaf region from complex backgrounds by modelling pixel intensity distributions. At first, (W_{pr}) is represented as a mixture of Gaussian distributions (G_{dt}).

$$G_{dt} = \sum v_t \delta(W_{pr} | \tau_t, c_t)$$

(3)

Where, v_t represents the weight of (W_{pr}), τ_t and c_t specify the mean and covariance values, respectively, and δ defines the normal distribution function. Then, the expectation function (E_{pt}) computes the responsibility of each data point using the current parameter values.

$$E_{pt} = \frac{v_t \delta(W_{pr} | \tau_t, c_t)}{\sum v_t \delta(W_{pr} | \tau_t, c_t)}$$

(4)

Then, in the maximization function, the parameters are updated.

$$(\tau_{t+1}, c_{t+1}, v_{t+1}) = \left(\frac{\sum E_{pt} W_{pr}}{N_p}, \frac{\sum E_{pt} (W_{pr} - \tau_t)^2}{N_p}, \frac{N_p}{\delta} \right)$$

(5)

Where, $\tau_{t+1}, c_{t+1},$ and v_{t+1} specify the updated mean, covariance, and weight values, respectively, and N_p is the effective number of points. Based on the points, the backgrounds are removed, and the obtained foreground leaf image is denoted as (λ_{fg}).

3.4 Vein structure extraction

Here, from (λ_{fg}), the details relevant to the vein of the leaf are extracted using the HGSSMM approach. The conventional Hessian Matrix Method (HMM) extracts veins by analyzing vein line curvature and orientation. However, its performance strongly depends on scale parameters, and improper selection leads to inaccurate extraction. Therefore, the proposed method uses the Gaussian Scale-Space (GSS) function to automatically select optimal parameters using a control function. Initially, (λ_{fg}) is fixed with the scale function based on GSS (ψ_s).

$$G_{sc} = \lambda_{fg}(x, y) \times \psi_s(x, y)$$

(6)

$$\psi_s(x, y) = \frac{1}{2\pi\rho^2} \exp\left(-\frac{x^2 + y^2}{2\rho^2} \right)$$

(7)

Here, G_{sc} defines the scale fixing input data, ρ is the scale parameter, $\exp$ defines the exponential function, and π is the numerical value. Then, the partial derivatives $\left(G_{sc}^{xx}, G_{sc}^{yy}, G_{sc}^{xy}\right)$ are calculated for $\left(G_{sc}\right)$.

$$\left(G_{sc}^{xx}, G_{sc}^{yy}, G_{sc}^{xy}\right) = \left(\frac{\partial^2 G_{sc}}{\partial x^2}, \frac{\partial^2 G_{sc}}{\partial y^2}, \frac{\partial^2 G_{sc}}{\partial x \partial y}\right)$$

(8)

Where, ∂ defines the derivative function. Then, the Hessian matrix (Z) is constructed based on the partial derivatives. After that, the eigenvalues are calculated based on the (Z). The veins are extracted from $\left(\lambda_{fg}\right)$, and the extracted vein structure is notated as $\left(V_s\right)$.

3.5 Color conversion

Meanwhile, Red, Green, Blue (RGB) of $\left(\lambda_{fg}\right)$ is converted into Hue, Saturation, Value (HSV) for extracting more intuitive color manipulation and structure changes of the leaf image. The converted image is represented as $\left(T_{fm}\right)$.

3.6 Feature extraction

In this section, from $\left(V_s\right)$, vein length, branching patterns, vein thickness, vein width, vein curvature, orientation, end points, junction points, branch length variance, and vein-to-leaf ratio features are extracted. Also, from $\left(T_{fm}\right)$, the Hue ratio, histogram of Hue values, saturation variance, low-saturation pixel ratio, mean brightness, Grey Level Co-occurrence Matrix (GLCM), Local Binary Pattern (LBP), edge, and Hu moments features are extracted. Overall, the extracted features are denoted as $\left(\zeta_{ef}\right)$.

3.7 Adaptive feature similarity boundary extraction

Here, from $\left(\zeta_{ef}\right)$, the adaptive feature similarity boundaries are extracted to differentiate subtle changes of different features at an early stage using the ABDNN approach. The conventional Approximate Nearest Neighbor (ANN) approach performs similarity matching in high-dimensional feature space using Euclidean Distance (ED). However, ED considers only the geometric distance between feature values. Therefore, this work uses Bhattacharyya Distance (BD), which effectively measures similarity between different probability distributions. Initially, two feature subsets, namely $\left(\varpi_{sk}\right)$ and $\left(\varpi_{sv}\right)$, are taken from $\left(\zeta_{ef}\right)$. Then, BD $\left(\beta_{df}\right)$ is calculated between these feature subsets.

$$\beta_{df} = -\ln\left(\sum \sqrt{\left(\varpi_{sk}\right)\left(\varpi_{sv}\right)}\right)$$

(9)

Where, $\ln$ refers to the log function. Then, all the data points are extracted and further checked based on the error function.

$$\beta_{df} \leq (1+\eta)$$

(10)

Here, η specifies the allowed approximation error. If the error is high, then the differences are recomputed, and similar points are grouped $\left(J_d\right)$. From $\left(J_d\right)$, the minimum points are selected.

$$\Re_{bd} = \arg\min\left(J_d\right)$$

(11)

Where, $\Re_{bd}$ defines the extracted adaptive feature similarity boundaries. The pseudocode of ABDNN is given below,

Pseudocode: ABDNN

Input: (ζ_{ef})

Output: $\Re_{bd}$

Begin

Initialize (ϖ_{sk}) and (ϖ_{sv})

For each (ζ_{ef}) **do**

Derive distance based on BD $\beta_{df} = -\ln\left(\sum\sqrt{(\varpi_{sk})(\varpi_{sv})}\right)$

Check the data point error

If $(\text{error} == \text{high})$
{Re-compute distance**} else**
{Obtained similar points **} end if**

Select Minimum point as final boundary

End for

Return $\Re_{bd}$
End

The extracted adaptive feature similarity helps learn the closest boundary.

3.8 Crop disease classification

In this section, $\Re_{bd}$, (ζ_{ef}), and (W_{pr}) are given as input to the proposed ETENSN approach for crop disease classification, and the combination of input data is declared as (τ_{ps}). The conventional EfficientNet effectively captures fine-grained features and preserves spatial information. However, its performance depends on proper coefficient scaling. Improper scaling may degrade the performance. Moreover, for large-scale training, unstable gradients may be presented. Therefore, this work applies Elastic Net Regularization (ENR) for optimal coefficient scaling and the Softplus activation function to smooth gradients for stable learning. Additionally, Transfer Learning (TL) is integrated to improve performance across diverse disease classes. The architecture diagram for the proposed ETENSN approach is given in Figure 2.

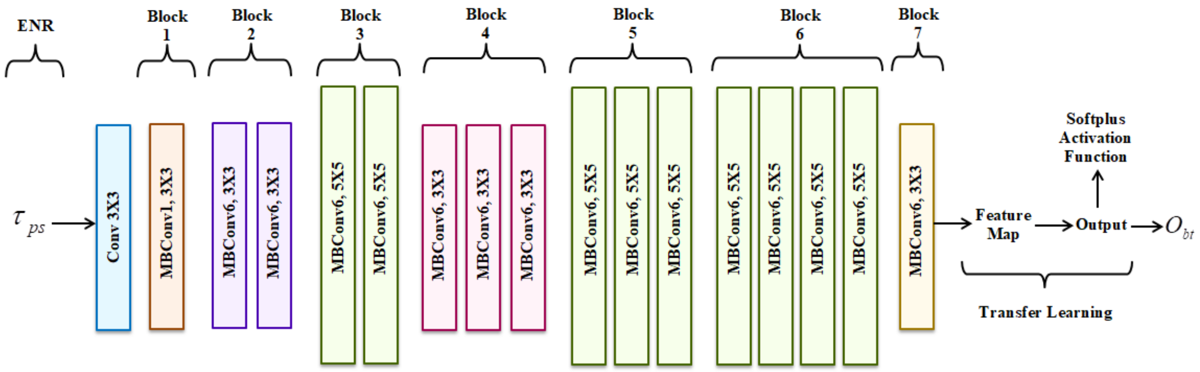


Figure 2: Architecture of the proposed ETENSN approach

Initially, coefficient scaling of parameters is initialized based on ENR.

$$\Im_{rc} = \min_{\alpha}\left\{\frac{1}{2r}\sum\left(\omega_{gt} - \tau_{ps}\alpha\right)^2 + \nu\left(\gamma\|\alpha\| + \frac{1-\gamma}{2}\|\alpha\|^2\right)\right\}$$

(12)

Where, $\Im_{rc}$ defines the regularized parameters, α is the model coefficients, ω_{gt} denotes the target value, r specifies the number of samples, ν is defined as the regularization strength parameter, and γ defines the mixing parameter. Next, (τ_{ps}) is given to the convolutional layer, which extracts the low-level features (χ_{low}).

$$\chi_{low} = \text{conv}_{3*3} * \tau_{ps}$$

(13)

Here, conv_{3*3} specifies the 3*3 convolution process. Then, the extracted (χ_{low}) is given to the Mobile inverted Bottleneck Convolution (MBConv) blocks that include expansion, depthwise convolution, and the squeeze and excitation. The squeeze and excitation block $(Q)$is used to learn channel-wise importance.

$$\chi_{hig} = u_{ws} \cdot \text{conv}_{1*1}(\chi_{low}) + Q$$

(14)

Where, χ_{hig} specifies the extracted high-level feature, u_{ws} is the depth-wise separable convolution, and conv_{1*1} indicates the 1*1 convolution process. Then, the global average pooling $(g_b)$is applied in χ_{hig} to obtain a feature vector(E_{vd}).

$$E_{vd} = g_b(\chi_{hig})$$

(15)

After that, the fully connected layer $(\zeta_{fc})$is used to calculate the class probability.

$$(\zeta_{fc}) = \mu_{sp} \cdot (\mathrm{E}_{vd} \times q_{wg}) + q_{ba}$$

(16)

$$\mu_{sp} = \ln(1 + \exp(E_{vd}))$$

(17)

Here, μ_{sp} is the softplus activation function, exp defines the exponential function, and q_{wg} and q_{ba} indicate the weight and bias value, which belong to $\Im_{rc}$, respectively. To adapt to the diverse disease cases, the proposed research work includes the TL for training the data. The pseudocode for the proposed ETENSN approach is given below,

Pseudocode: ETENSN

Input: (τ_{ps})

Output: (o_{bt})

Begin

Initialize ν , ω_{gt} and γ

For each (Z_{ed}) **do**

Parameter coefficient scaling of parameters by $\Im_{rc}$

Derive MBConv operation base on (Q)

Perform global average pooling

Derive (ζ_{fc}) based on $\mu_{sp} = \ln(1+\exp(E_{vd}))$

Average the feature map G_{avg}

Trained the model based on TL

If $(\text{loss} == \text{minimum})$ **{Terminate } else**
{Repeat } end if

End for

Return (o_{bt})
End

At last, the obtained outcome is declared as (o_{bt}), which consists of both healthy (d_{he}) and disease classes (d_{es}).

3.9 Nutrient recommendation

Here, for the predicted diseases (d_{es}), the nutrient is recommended to further preserve the crop from disease using the FTMIS approach. The conventional Fuzzy Inference System (FIS) effectively handles nonlinear systems and complex environments. But, its decision-making relies only on the maximum membership value, ignoring contributions from other data points and causing information loss. Therefore, this work employs Tilted Maxout (TM) point calculation, which considers all data points using a confidence distance function for improved decision accuracy. At first, the input data is fuzzified based on the Gaussian Membership (GM) (ι_{mem}) function.

$$\iota_{mem} = \exp\left(-\frac{(d_{es} - c_{en})^2}{2\varphi^2}\right)$$

(18)

Here, c_{en} and φ denote the center and width values, respectively. After that, the rule (λ_{rl}) is generated.

$$\lambda_{rl} = \begin{cases} P_t, Ca, & \text{if } (d_{es} = L_b \text{(or)} T_{mv} \text{(or)} L_m) \\ P_t, P_r, & \text{if } (d_{es} = E_{bl} \text{(or)} S_{mls} \text{(or)} T_s) \\ Ca, S_f, Cu, & \text{if } (d_{es} = B_{st} \text{(or)} P_{md}) \\ P_t, Mg, & \text{if } (d_{es} = T_{yl}) \\ Ni, Mg, S_f, & \text{if } (d_{es} = S_{mi}) \end{cases}$$

(19)

Where, P_t, Ca, P_r, S_f, Cu, Mg, and Ni define the potassium, calcium, phosphorus, sulphur, copper, magnesium, and nitrogen, respectively. $L_b, T_{mv}, L_m, E_{bl}, S_{mls}, T_s$, B_{st}, P_{md}, T_{yl}, and S_{mi} denote the late blight, tomato mosaic virus, leaf mold, early blight, septoria leaf spot, target spot, bacterial spot, powdery mildew, tomato yellow leaf curl virus, and spider mites, respectively. Then, the inference engine (I_{ef}) is calculated based on the (λ_{rl}) and (ι_{mem}). Next, the defuzzification (D_{fc}) process is carried out based on TM.

$$D_{fc} = \arg\max(\iota_{mem} + \ell_{ti} \cdot \wp_{cf})$$

(20)

Where, ℓ_{ti} denotes the tilt controlling parameter and $\wp_{cf}$ is the confidence distance between (I_{ef}) of disease classes. The experimental assessment is explained in a further section.

4. RESULT AND DISCUSSION

In this section, the performance of the proposed tomato crop disease monitoring system is evaluated. The proposed work is implemented in the working platform of Python.

4.1 Dataset description

For the performance assessment, this research work utilizes the "Tomato Leaves Dataset" collected from the Kaggle source, and the source link of the dataset is given in the reference section. Details of the dataset are given in Table 1.

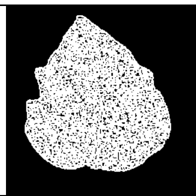
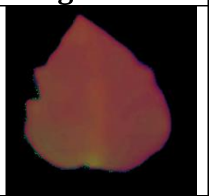
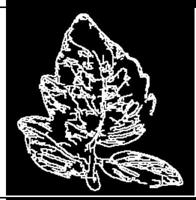
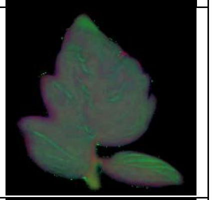
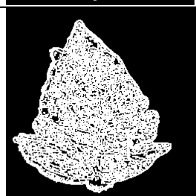
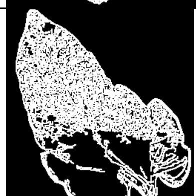
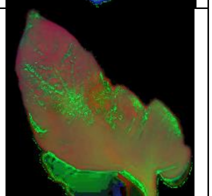
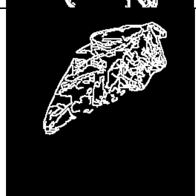
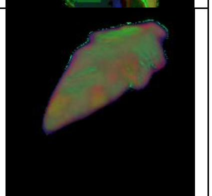
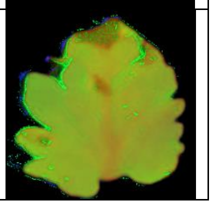
Table 1: Details of the dataset

S. No	Classes	Image count
1	Healthy	3051
2	Bacterial spot	2826
3	Early blight	2455
4	Late blight	3113
5	Leaf mold	2754
6	Septoria leaf spot	2882
7	Spider mites, two-spotted spider mite	1747

8	Target spot	1827
9	Tomato yellow leaf curl virus	2039
10	Tomato mosaic virus	2153
11	Powdery mildew	1004
Total		25851

After data balancing, each class consists of 3113 images, and the total count is 34243. Among the total count, 80% (i.e., 27394) of the images are used for training and the remaining 20% (i.e., 6849) of the images are used for testing purposes.

Table 2: Sample image results

Classes/ Phases	**Input**	**Noise-removed image**	**Background removed image**	**Vein structure extraction**	**RGB to HSV converted image**
Healthy					
Bacterial spot					
Early blight					
Late blight					
Leaf mold					
Septoria leaf spot					

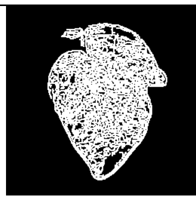
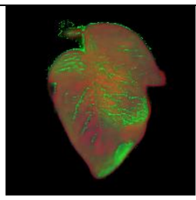
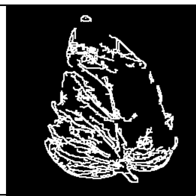
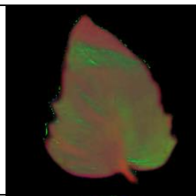
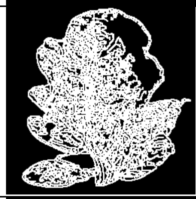
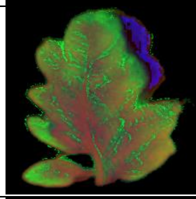
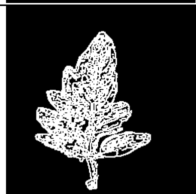
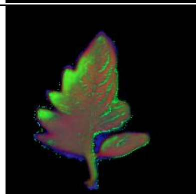
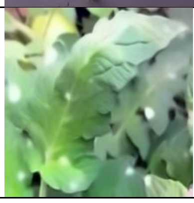
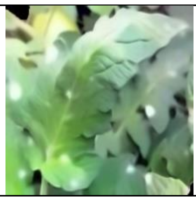
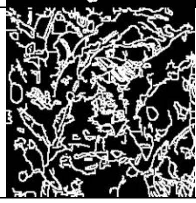
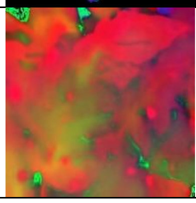
Spider mites, two-spotted spider mite					
Target spot					
Tomato yellow leaf curl virus					
Tomato mosaic virus					
Powdery mildew					

Table 2 shows the sample image results, such as the input, noise-removed image, background-removed image, vein structure extracted image, and color converted image.

4.2 Performance assessment

Here, the performance of the proposed and baseline models is analyzed.

Table 3: Quantitative metric-based performance assessment for feature similarity extraction

Methods/Metrics	NDE	ICSS	BSI
Proposed ABDNN	0.082	0.941	0.918
ANN	0.117	0.903	0.874
KNN	0.163	0.856	0.821
RPM	0.214	0.802	0.768
LR	0.267	0.741	0.703

Table 3 shows the quantitative metric-based performance validation of the proposed ABDNN and existing ANN, K-Nearest Neighbor (KNN), Random Projection Method (RPM), and Linear Regression (LR) approaches. Due to the usage of BD, the proposed ABDNN approach attained 0.082 Neighbor Distance Error (NDE), 0.941 Inter-Class Similarity Score (ICSS), and 0.918 Boundary Stability Index (BSI). In contrast, prevailing methods attained higher NDE and lower ICSS and BSI.

Table 4: Performance assessment of crop disease classification

Methods/Metrics	Accuracy (%)	Precision (%)	Recall (%)	F-Measure (%)	FPR	FNR
Proposed ETENSN	99.6235	99.4577	99.3699	99.4138	0.0655	0.0701
EfficientNet	96.4771	96.0214	96.0147	96.018	0.0857	0.0936
DenseNet	90.2314	89.9654	89.8624	89.9139	0.1745	0.1896
ResNet	86.3145	86.1788	86.0336	86.1062	1.2647	1.3224
Inception-v3	81.6745	81.2479	81.1476	81.1977	1.6428	1.7598

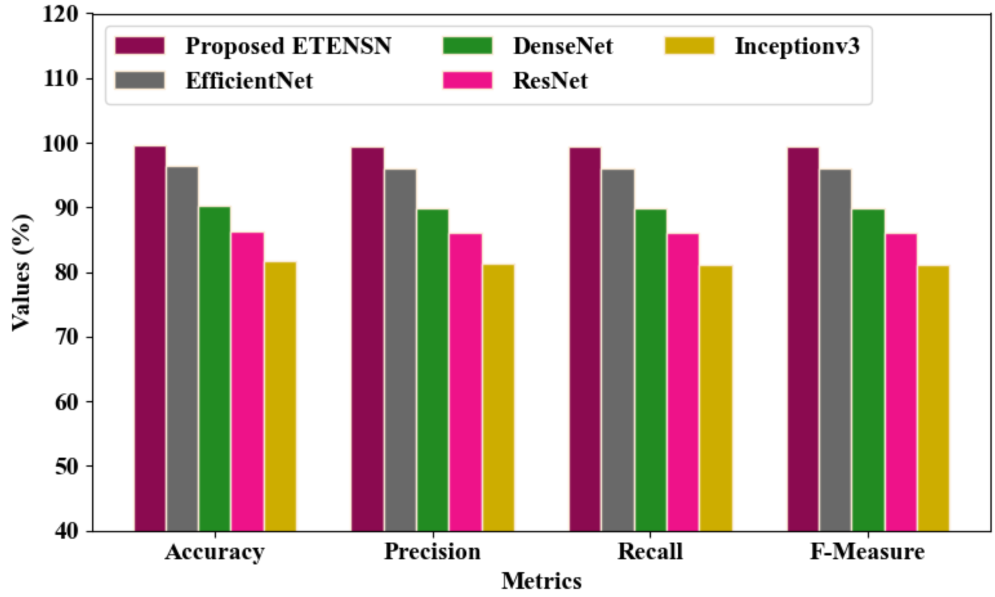


Figure 3: Pictorial plot for performance assessment of crop disease classification

Table 4 and Figure 3 show the performance of the proposed ETENSN approach and existing EfficientNet, Densely connected convolutional Network (DenseNet), Residual Network (ResNet), and Inception-version 3 (Inception-v3). Here, owing to the usage of ENR, softplus, and TL, the proposed ETENSN attained 99.6235% accuracy, 99.4577% precision, 99.3699% recall, 99.4138% F-measure, 0.0655 False Positive Rate (FPR), and 0.0701 False Negative Rate (FNR) values. In contrast, the average accuracy value of the existing approach was 88.6743%, which was lower, and based on other metrics, the existing models also provided poor performance. This showed that the proposed ETENSN approach was highly supported for a real-time crop monitoring system.

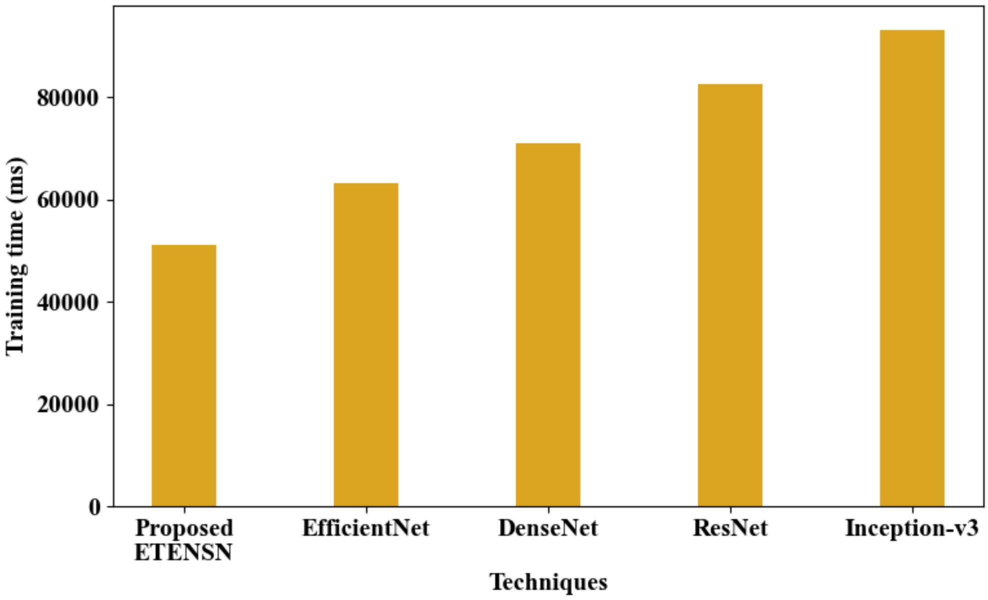


Figure 4: Training time analysis

Training time of the proposed and existing approach-based crop disease classification is shown in Figure 4. Here, the proposed ETENSN took 51234ms training time; yet, the average training time of the existing approach was 77523ms, which was higher than the proposed approach.

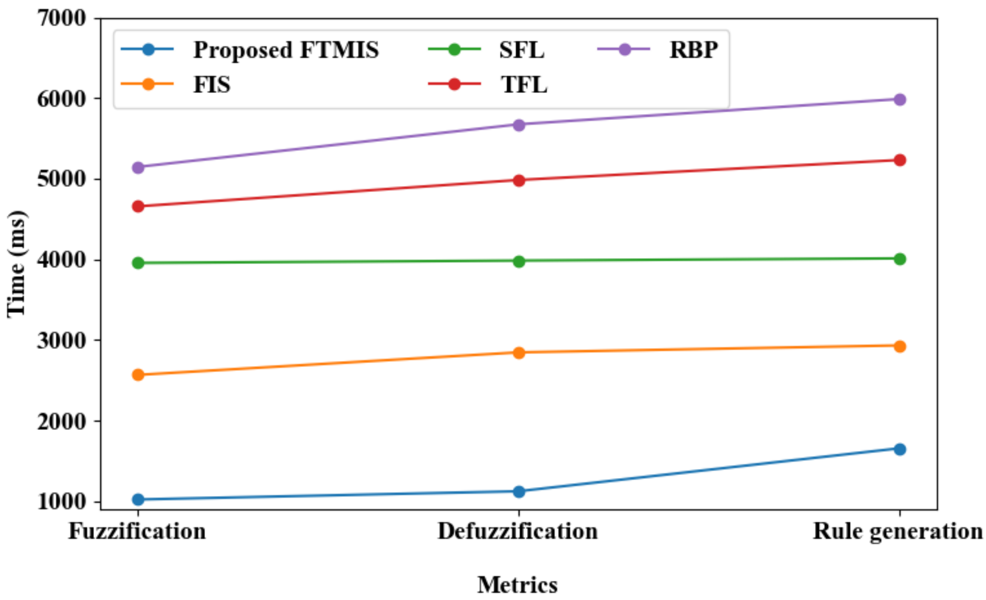
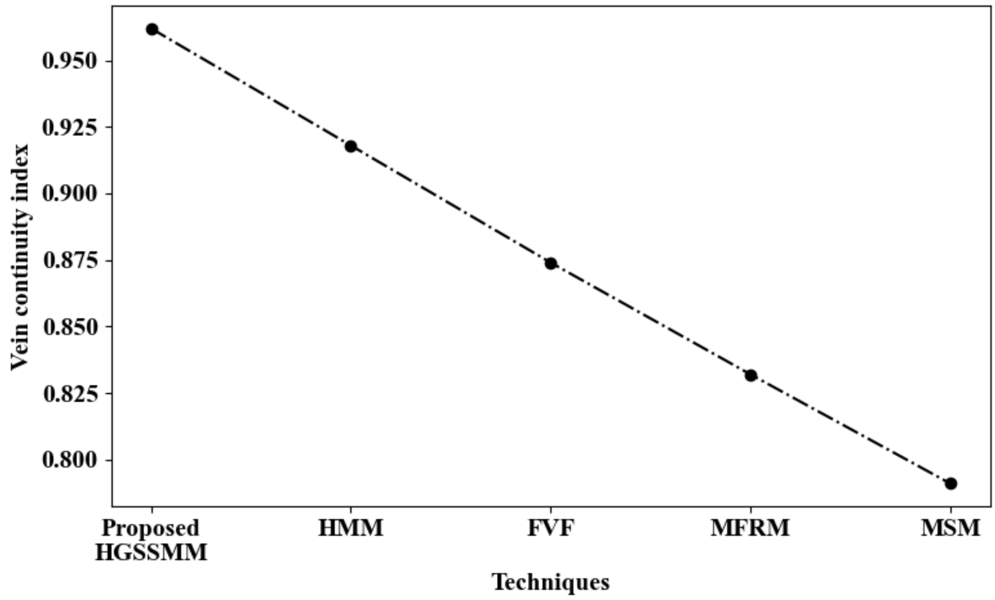
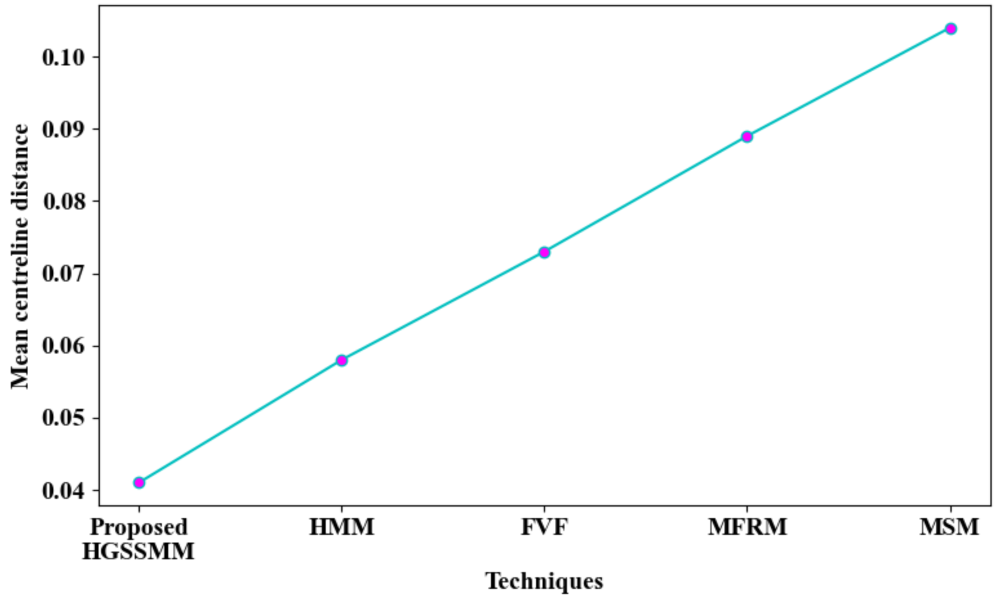


Figure 5: Nutrient recommendation performance assessment

Figure 5 displays the nutrient recommendation performance of the proposed FTMIS and existing FIS, Sigmoid Fuzzy Logic (SFL), Trapezoidal Fuzzy Logic (TFL), and Rule-Based Prediction (RBP). The fuzzification, defuzzification, and rule generation times of the proposed approach were 1023ms, 1125ms, and 1657ms, respectively, which were lower than those of the existing approaches because of the usage of TM. But, the prevailing approaches took more time.



(a)



(b)

Figure 6: Graphical plot for the performance evaluation of vein structure extraction

Figure 6 shows the graphical representation of the performance of the proposed HGSSMM and existing HMM, Frangi Vesselness Filter (FVF), Matched Filter Response Method (MFRM), and Morphological Segmentation Method (MSM) regarding vein structure extraction. The vein continuity index and mean centreline distance of the proposed HGSSMM approach were 0.962 and 0.041, respectively. But, the prevailing models attained a lower vein continuity index and a higher mean centreline distance than the proposed approach. Due to the usage of GSS, the proposed approach attained better performance.

4.3 Comparative analysis

Table 5: Comparative analysis between proposed and state-of-the-art works

Author's name	**Objective**	**Method used**	**Accuracy**
(Sakkarvarthi et al., 2022)	Detection and Classification of Tomato Crop Disease	CNN	98%
(Trivedi et al., 2021)	Early Detection and Classification of Tomato Leaf Disease	High-performance Deep neural network	98.49%
(Mac et al., 2024)	Intelligent agricultural robotic detection system	ResNet-152	92.32%
(Beyza et al., 2024)	Classification of Diseases in Tomato Leaves	DL methods	96%
(Khan et al., 2024)	Automated Tomato Leaf Disease Detection Using Image Processing	SVM	92.3%
Proposed	Mobile tomato crop disease monitoring for precision farming	ETENSN	99.6235%

Table 5 shows the performance of the proposed model and the state-of-the-art works based on accuracy metrics. Due to the inclusion of adaptive feature similarity boundary extraction and vein details extraction, the proposed ETENSN-approach-based mobile tomato crop disease monitoring system attained 99.6235%

accuracy, which was higher than the existing works. Existing CNN and High-performance deep neural network-based models attained higher performance compared with the other prevailing works; yet, those models attained lower performance compared with the proposed framework.

5. CONCLUSION

This paper presented the ETENSN approach-based tomato crop disease detection. Initially, noises were removed, and then, the background of the image was removed using GMM. Next, the proposed HGSSMM approach-based vein structure extraction attained a 0.962 vein continuity index, which was higher than the existing methods. Then, the ABDNN approach-based adaptive feature similarity boundary extraction attained a lower NDE and higher ICSS and BSI. At last, the proposed ETENSN approach-based crop disease classification achieved 99.6235% accuracy. Then, the nutrient recommendation using the proposed FTMIS took the lowest fuzzification, defuzzification, and rule generation times. Overall, the proposed framework provided better results than the baseline models and the state-of-the-art-works.

Practical implication: The proposed framework enables real-time mobile tomato disease detection for farmers and supports accurate nutrient-based crop management. Moreover, it reduces yield loss and pesticide misuse.

Future scope: In the future, the proposed research work will be extended by interpreting decision-making to improve the transparency of the decision.

REFERENCES:

1. Dataset Link: https://www.kaggle.com/datasets/ashishmotwani/tomato
2. Al-Shamasneh, A. R., & Ibrahim, R. W. (2024). Classification of tomato leaf images for detection of plant disease using conformable polynomials image features. *MethodsX*, *13*, 1–13. https://doi.org/10.1016/j.mex.2024.102844
3. Alzahrani, M. S., & Alsaade, F. W. (2023). Transform and Deep Learning Algorithms for the Early Detection and Recognition of Tomato Leaf Disease. *Agronomy*, *13*, 1–24. https://doi.org/10.3390/agronomy13051184
4. Beyza, M., Hafif, M. F., & Koksoy, E. K. (2024). Classification of Diseases in Tomato Leaves Using Deep Learning Methods. *INTELLIGENT METHODS in Engineering Science*, *3*(1), 22–36. https://imiens.org/index.php/imiens/article/download/48/28
5. Chen, M., Wang, C., Liu, C., Yu, Y., Yuan, Y., Ma, J., & Zhang, K. (2025). Tomato leaf disease detection method based on improved YOLOv8n. *Scientific Reports*, *15*, 1–21. https://doi.org/10.1038/s41598-025-00405-8
6. Deng, Y., Xi, H., Zhou, G., Chen, A., Wang, Y., Li, L., & Hu, Y. (2023). An Effective Image-Based Tomato Leaf Disease Segmentation Method Using MC-UNet. *Plant Phenomics*, *5*, 1–17. https://doi.org/10.34133/plantphenomics.0049
7. Ghosh, H., Sadiq, I., Mintajur, R., Emon, R., Mashrafi, J., Tanzin, M. A. A. A., Mohanty, S. N., & Kant, S. (2025). Advanced neural network architectures for tomato leaf disease diagnosis in precision agriculture. *Discover Sustainability*, *6*, 1–30. https://doi.org/10.1007/s43621-025-01149-1
8. Harakannanavar, S. S., Rudagi, J. M., Puranikmath, V. I., Siddiqua, A., & Pramodhini, R. (2022). Plant leaf disease detection using computer vision and machine learning algorithms. *Global Transitions Proceedings*, *3*, 305–310. https://doi.org/10.1016/j.gltp.2022.03.016
9. Hossain, I., Jahan, S., Asif, R. Al, Samsuddoha, & Ahmed, K. (2023). Detecting tomato leaf diseases by image processing through deep convolutional neural networks. *Smart Agricultural Technology*, *5*, 1–10. https://doi.org/10.1016/j.atech.2023.100301
10. Khan, R., Din, N. U., Zaman, A., & Huang, B. (2024). Automated Tomato Leaf Disease Detection Using Image Processing: An SVM-Based Approach with GLCM and SIFT Features. *Journal of Engineering*, *2024*, 1–10. https://doi.org/10.1155/2024/9918296

11. Khasawneh, N., Faouri, E., & Fraiwan, M. (2022). Automatic Detection of Tomato Diseases Using Deep Transfer Learning. *Applied Sciences*, *12*, 1–17. https://doi.org/10.3390/app12178467
12. Mac, T. T., Nguyen, T., Dang, H., Nguyen, D., & Nguyen, X. (2024). Intelligent agricultural robotic detection system for greenhouse tomato leaf diseases using soft computing techniques and deep learning. *Scientific Reports*, *14*, 1–22. https://doi.org/10.1038/s41598-024-75285-5
13. Nawaz, M., Nazir, T., Javed, A., Masood, M., Rashid, J., Kim, J., & Hussain, A. (2022). A robust deep learning approach for tomato plant leaf disease localization and classification. *Scientific Reports*, *12*, 1–18. https://doi.org/10.1038/s41598-022-21498-5
14. Nguyen, H. T., Huong Hoang Luong, Huynh, L. B., Le, B. Q. H., Doan, N. H., & Le, D. T. D. (2023). An Improved MobileNet for Disease Detection on Tomato Leaves. *Advances in Technology Innovation*, *8*(3), 192–209. https://www.academia.edu/download/124879029/1398.pdf
15. Oni, M. K., & Prama, T. T. (2025). Optimized Custom CNN for Real-Time Tomato Leaf Disease Detection. *Arxiv*, 1–15. https://arxiv.org/pdf/2502.18521?
16. Ruhad, F. M., Fahim, M., Hossain, M. S., Monir, M. F., Islam, A., & Amin, M. A. (2025). Beyond Classification: Benchmarking Object Detection Models for Efficient Tomato Leaf Disease Identification on a Real-World Dataset. *Smart Agricultural Technology*, 1-15. https://doi.org/10.1016/j.atech.2025.101336
17. Sakkarvarthi, G., Sathianesan, G. W., Murugan, V. S., Reddy, A. J., Jayagopal, P., & Elsisi, M. (2022). Detection and Classification of Tomato Crop Disease Using Convolutional Neural Network. *Electronics*, *11*, 1–13. https://doi.org/10.3390/electronics11213618
18. Tarek, H., Aly, H., Eisa, S., & Abul-Soud, M. (2022). Optimized Deep Learning Algorithms for Tomato Leaf Disease Detection with Hardware Deployment. *Electronics*, *11*, 1–19. https://doi.org/10.3390/electronics11010140
19. Trivedi, N. K., Gautam, V., Anand, A., Aljahdali, H. M., Villar, S. G., Anand, D., Goyal, N., & Kadry, S. (2021). Early Detection and Classification of Tomato Leaf Disease Using High-Performance Deep Neural Network. *Sensors*, *21*, 1–15. https://doi.org/10.3390/s21237987
20. Yu, A., Xiong, Y., Lv, Z., Wang, P., She, J., & Wei, L. (2024). New Method for Tomato Disease Detection Based on Image Segmentation and Cycle-GAN Enhancement. *Sensors*, *24*, 1–17. https://doi.org/10.3390/s24206692
21. Zhang, E., Zhang, N., Li, F., & Lv, C. (2024). A lightweight dual-attention network for tomato leaf disease identification. *Frontiers in Plant Science*, *15*, 1–18. https://doi.org/10.3389/fpls.2024.1420584