

Anomaly Detection in Hydroponic Maize Fodder Through Image Processing

Christian Yair López Martínez¹ ; Jesús Alberto González Pedroza¹ ;

Dana Paola Feria Torres¹ ; Laura Ivoone Garay Jiménez¹ 

¹Instituto Politécnico Nacional, UPIITA, México,
lopezchristian992@gmail.com, albertoglezjp@gmail.com,
danaferia02@gmail.com, lgaray@ipn.mx

Abstract— This work presents a specialized automated monitoring system for hydroponic maize green fodder that addresses key limitations of existing plant disease detection approaches. Unlike prior works focusing on field crops with high-end computing infrastructure, our system operates on low-cost ESP32-CAM hardware while achieving competitive accuracy. The system integrates classical HSV-based segmentation with MobileNetV2 classification, exploiting multiple camera viewpoints (superior and inferior) for enhanced diagnostic reliability. Fixed-size patches (250×250 pixels) extracted from segmented regions serve as input to the binary classifier (healthy/diseased). The network achieved 95.01% accuracy for superior view and 97.28% for inferior view, comparable to state-of-the-art approaches using computationally expensive architectures. Key contributions include: (1) specialized preprocessing for hydroponic imaging conditions, (2) edge-compatible deployment maintaining high accuracy, (3) integrated height measurement for comprehensive crop monitoring, and (4) multi-view assessment enhancing diagnostic confidence. The system demonstrates the feasibility of deploying efficient deep learning solutions in resource-constrained agricultural environments.

Keywords— Hydroponic maize fodder, image processing, anomaly detection, deep learning, MobileNetV2.

I. INTRODUCTION

Hydroponics is a cultivation method that dispenses with soil, utilising nutrient solutions dissolved in water to supply plants with the essential elements for their growth. This method allows for greater control over crop development conditions, optimising the use of water and nutrients. Different types of hydroponics exist, such as the nutrient film technique (NFT); the floating root technique; and the inert substrate technique [1]. This method is particularly useful in regions with scarce fertile soil or adverse climatic conditions, as it reduces dependence on environmental factors and enables crop production in closed and/or controlled spaces. However, its success depends on constant supervision, since any alteration in water quality or nutrient composition significantly affects crop development. Hence the importance of early diagnosis of diseases in crops.

The application of deep learning techniques in agriculture has experienced significant growth in recent years, demonstrating remarkable capabilities for automated plant disease detection and crop monitoring [2], [3]. Convolutional neural

networks (CNNs) have proven particularly effective in extracting relevant features from plant images, enabling accurate classification of healthy and diseased specimens [4]. Recent comprehensive reviews highlight the evolution of deep learning models specifically designed for plant disease detection, with reported accuracies ranging from 85% to 99% depending on crop type, disease complexity, and imaging conditions [5], [6].

Despite these advances, most existing works focus on soil-based agriculture and outdoor crops, with limited attention to controlled hydroponic environments. Furthermore, existing systems typically require high-end computing infrastructure, limiting their deployment in resource-constrained agricultural settings. MobileNet architectures have emerged as promising solutions for edge deployment in agricultural applications [7], achieving competitive accuracies (92–96%) while maintaining computational efficiency suitable for embedded systems [8].

This research addresses these gaps by developing a specialized monitoring system for hydroponic maize green fodder that operates on low-cost ESP32-CAM hardware. The system integrates classical image processing with efficient deep learning architectures to enable real-time disease detection and growth monitoring in controlled environments. The proposed approach combines HSV-based segmentation with MobileNetV2 classification, achieving high accuracy (95.01% superior view, 97.28% inferior view) while maintaining compatibility with resource-constrained devices. The integration of multiple camera viewpoints and simultaneous height measurement represents a novel contribution for comprehensive crop assessment in hydroponic systems [9], [10].

II. RELATED WORK

Recent advances in plant disease detection have demonstrated the effectiveness of deep learning approaches across various crops and imaging conditions. Table I summarizes representative works in this domain, highlighting their methodologies, target crops, and performance metrics.

While existing approaches achieve high accuracies, they present three main limitations for hydroponic applications: (1) most target field crops rather than controlled hydroponic systems, (2) they require high-computational resources incompatible with edge deployment, and (3) they focus solely on disease classification without integrating growth monitoring. Our

TABLE I
Comparison of Deep Learning Approaches for Plant Disease Detection

Reference	Crop / Environ.	Architect.	Acc.	Edge Dev.
Upadhyay et al. [5]	Multiple crops (field)	ResNet-50	96.2%	No
Ashurov et al. [10]	Tomato, pepper (field)	Depthwise CNN	94.5%	No
Shoaib et al. [6]	Various (field)	VGG-16	93.8%	No
Shoaib et al. [9]	Multiple (greenh.)	DenseNet	95.3%	No
This work	Maize fodder (hydrop.)	MobileNetV2	95.01%	Yes (ESP32)

work addresses these gaps by developing a system specifically designed for hydroponic environments that operates on low-cost embedded hardware (ESP32-CAM) while providing both disease detection and height measurement capabilities. The use of MobileNetV2 enables competitive accuracy (95.01%) with significantly reduced computational requirements compared to ResNet, VGG, or DenseNet architectures, making real-time processing feasible on resource-constrained devices.

III. METHODOLOGY

A. *Image Acquisition* For image acquisition, ESP32-CAM development boards equipped with the OV5640 optical sensor were utilised. These boards feature a basic system for image acquisition, digitalisation and transmission. The cameras were positioned at three different locations, seeking to minimise the effects of the following alterations:

1) *Considerations during image acquisition* During the capture stage, various conditions were identified that affected image quality. The most relevant factors are enumerated below:

- **Reflections:** Caused by the light source and the material used in the trays and bases of the station.
- **Shot distance:** According to the specifications of the ESP32-CAM optical sensor (OV5640), it was necessary to characterise the optimal distance.
- **Water:** Residual droplets from the irrigation system caused reflections that also affected shot quality.
- **Shadows:** The hydroponic system had a single illumination point, which generated shadows cast by the developing seedlings and root systems, perceived as black spots in the images.
- **Illumination variation:** The irregular morphology of the green fodder caused zones that were very well illuminated (“protagonist” points) and others with reduced illumination (“antagonist” points).

2) *Camera positioning* Under the conditions described above and considering the viewing angle of the cameras, the selected camera positions were defined and labelled as:

- **Shot 1 (T1):** Superior view. This shot encompasses the upper part of the green fodder, positioned at a distance of 23 cm from the translucent base to the camera support. It serves to perform green fodder classification.

- **Shot 2 (T2):** Frontal view. This shot captures the frontal part of the green fodder, with the camera positioned 13 cm from the front of the hydroponic prototype and the tray inclined at 8°.
- **Shot 3 (T3):** Inferior view. This shot captures the bottom of the tray containing the green fodder, positioned at a distance of 12 cm from the translucent base of the prototype. This view is also used for fodder classification.

Fig. 1 shows the assembled prototype with the camera positions.

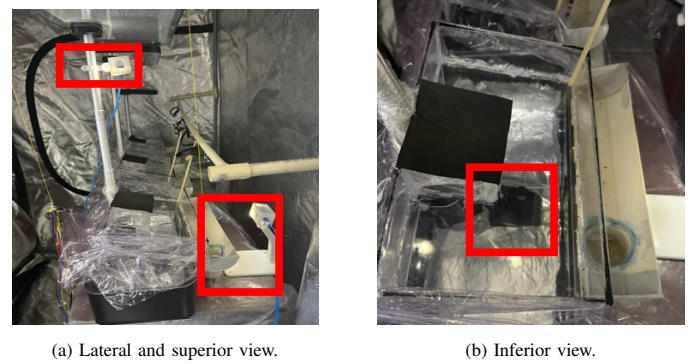


Fig. 1. Camera positioning.

As shown in Fig. 2, the original images required preprocessing to address suboptimal lighting conditions at the fodder production site during initial testing.

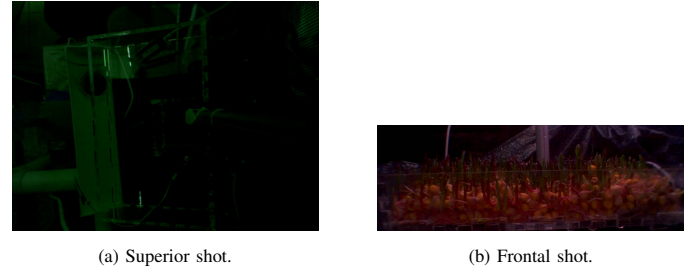


Fig. 2. Original images captured at the fodder production site.

3) *Camera operation* For camera operation, a sequence was performed in which adjustments were considered for the modification of parameters such as brightness, exposure, contrast and sensor gain, the calculation for the next shot and hibernation of the same.

4) *Camera configuration* Parameter adjustments were made, such as exposure, gain, brightness, contrast and image quality from the sensor, in order to characterise the image and extract the information of interest. Fig. 4 presents the implementation in the development board for the continuous monitoring process.

In this way, the transition was made from the information contained in Fig. 2a to the first characterised image shown in Fig. 5.

B. *Image Preprocessing* Preprocessing of the captured images is a critical step to guarantee the quality of input data and

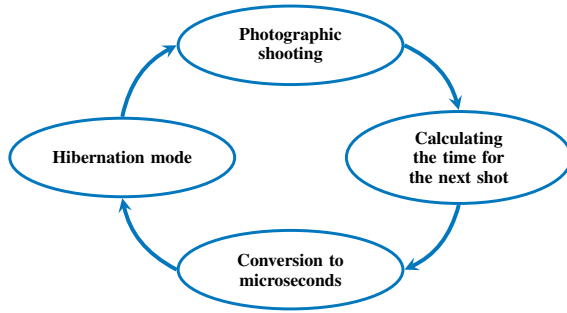


Fig. 3. Diagram of the image acquisition in the monitoring process.

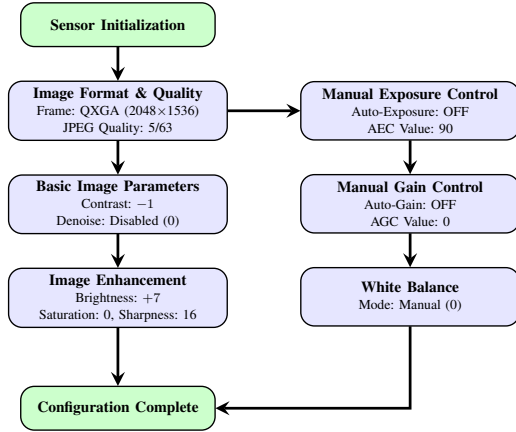


Fig. 4. OV5640 sensor configuration workflow during ESP32-CAM firmware initialization. The sensor is configured with manual exposure and gain control parameters (disabling automatic adjustments) to ensure consistent image capture conditions. All parameters are set once during system boot and remain fixed during operation.

improve the robustness of analysis algorithms (for example, green fodder height measurement, classification). First, three metrics are calculated (brightness, sharpness, noise) and, according to the result, it is decided automatically whether it is necessary to illuminate the scene, attenuate overexposure or intensify colours. The functions implemented for these measurements are shown in Fig. 6.

1) *Light measurement* Converts the image to HSV and averages the V channel; a single value is obtained that indicates how much light there is, which serves to choose between brightening, darkening or leaving the illumination as it is.

2) *Sharpness measurement* Converts RGB images to grayscale and applies edge detection filtering to compute a sharpness metric based on intensity variance. High variance values indicate a focused image with well-defined edges, while low variance values suggest a blurred image. This metric determines whether sharpening enhancement should be applied during preprocessing.

3) *Noise measurement* Calculates the variance of light intensities in each block and then averages all those variances to obtain a single value that represents the noise level present; if that average is high, the pipeline applies a median filter before any enhancement to smooth the image, whilst low values allow

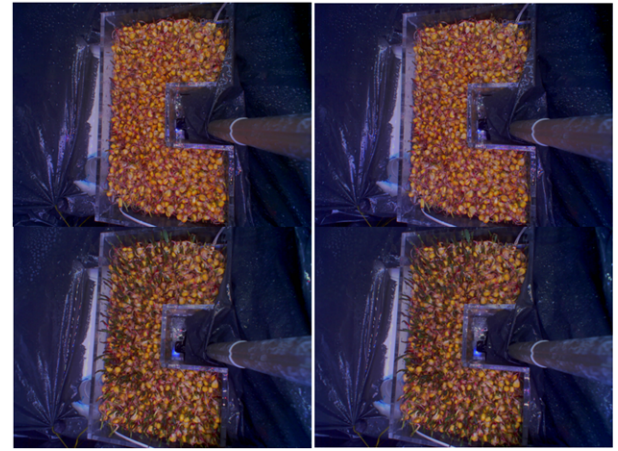


Fig. 5. Examples of the images obtained for the database, superior shot.

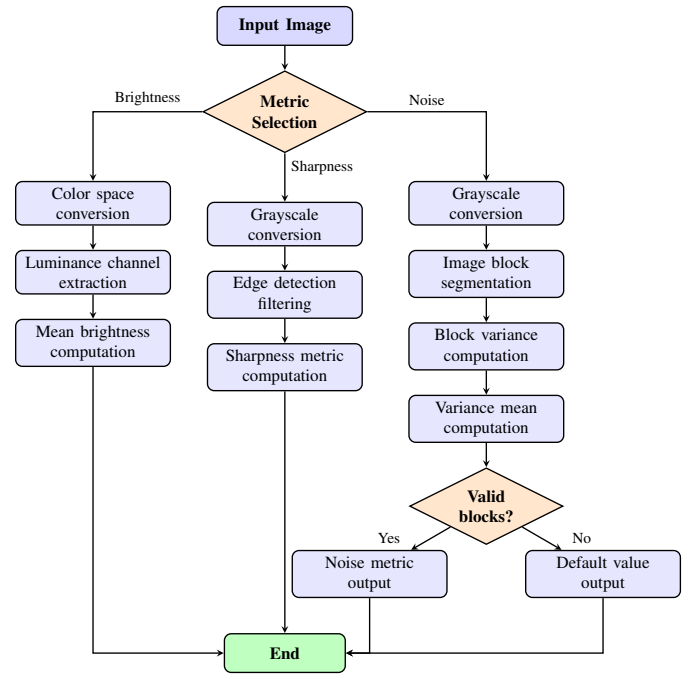


Fig. 6. Flowchart depicting the preprocessing functions for brightness, sharpness, and noise measurement with dynamic adaptation in preprocessing.

the smoothing to be omitted and more detail to be preserved.

4) *Adaptive histogram equalisation (CLAHE) in V channel* Contrast Limited Adaptive Histogram Equalization (CLAHE) [11] is applied exclusively to the luminance channel (V in HSV color space) to enhance local contrast and extract shadow details without modifying color information or amplifying noise artifacts. This approach is particularly effective for scenes with poor illumination conditions.

5) *Global luminance adjustment through gamma correction* Generates a lookup table (LUT) containing pre-computed gamma correction values for each pixel intensity level and applies it using `cv2.LUT`; brightens if $\gamma < 1$ or darkens if $\gamma > 1$, providing smooth global adjustment when the photo is only slightly unbalanced.

6) *Automatic gamma correction for overexposed scenes* Automatically calculates the gamma value necessary for the mean brightness to be situated at a target value and calls the previous function; it is used to correct overexposure precisely and without iterative adjustments [12].

7) *Saturation enhancement* Increases the saturation of the S channel (from the HSV color space) and clips the result to the permitted range; this enhances the greens of the plant, which later facilitates separating the crop from the background.

8) *Median filter* Applies a median filter of 3×3 , sufficient to eliminate noise points without erasing edges [12]; it is activated only when the estimated noise level exceeds a predefined threshold.

9) *Sharpening mask* Applies an unsharp masking technique: a Gaussian-blurred version of the image is subtracted from the original to create a high-pass filtered result that emphasizes edges [12]. The enhancement is selectively applied only in regions where significant edge information is detected, thus sharpness is improved only if really necessary and without amplifying residual noise in uniform areas.

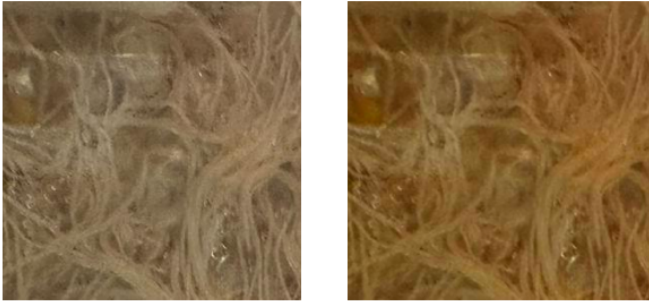


Fig. 7. Comparison between the original image (left) and the preprocessed image (right).

C. Image Processing

1) *RGB thresholds and patch size* In this section, the minimum and maximum ranges for each R, G and B channel are defined with which the mask of the region of interest will be created, in addition to the fixed size of each patch (`tile_size`). The structural elements for subsequent morphological operations are also defined. The parameters and configurations are shown in Fig. 8.

2) *Mask creation by thresholds* Subsequently, the three channels of the RGB image are extracted and a Boolean mask is constructed where each pixel meets the ranges defined in RGB Thresholds. The mask creation process is detailed in Fig. 9.

3) *Initial mask cleaning: opening, closing and hole filling* This part applies an opening with a small element to eliminate isolated small regions, then a closing and hole filling to join minimal internal areas, ensuring that the initial mask is more coherent before closing major ruptures. The morphological operations applied are shown in Fig. 10.

4) *Combination with the active region after major closing* A major closing operation with a large structural element

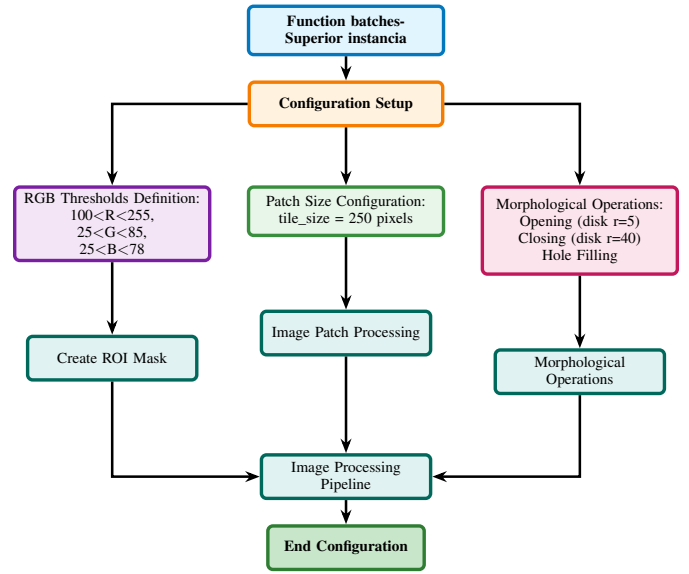


Fig. 8. Configuration flowchart showing RGB thresholds, patch size, and morphological operations for image processing.

(disk radius 40) is applied to connect regions separated by significant gaps. The resulting mask (`mask_closed`) is then combined with the previously filled mask (`mask_filled`) using a logical AND operation: pixels are retained only if they are `True` in both masks. This ensures that only sufficiently connected regions remain in the final combined mask (`mask_combined`). The mask combination process is shown in Fig. 11.

5) *Main region detection and bounding box* Once the combined mask is obtained, the region with greatest area is identified through connected component labelling. The properties of each region are extracted and that with the greatest area is selected, subsequently calculating the bounding box parameters that delimit the region of interest: top-left corner coordinates (x_0, y_0), width, height, and bottom-right corner coordinates. The detection process and bounding box calculation are detailed in Fig. 12.

6) *Patch traversal, index omission and saving* From the original image, several 250×250 pixel subimages were obtained and grouped as batches, which were used as input for the deep network. The purpose of this segmentation is to obtain different samples of the forage images and increase the quantity of samples for network training.

Counters are initialized to track the number of valid patches and control the extraction process. For each position in the sliding window grid (y_s, x_s), the corresponding submask region is extracted. If it contains at least one `True` pixel (indicating presence of fodder), the patch counter is incremented. If the counter value coincides with any of the predefined indices to be omitted according to laboratory criteria (e.g., patches containing tray edges or artifacts), the patch is skipped. Otherwise, the 250×250 pixel patch is extracted from the RGB image, converted to BGR format for compatibility, encoded as

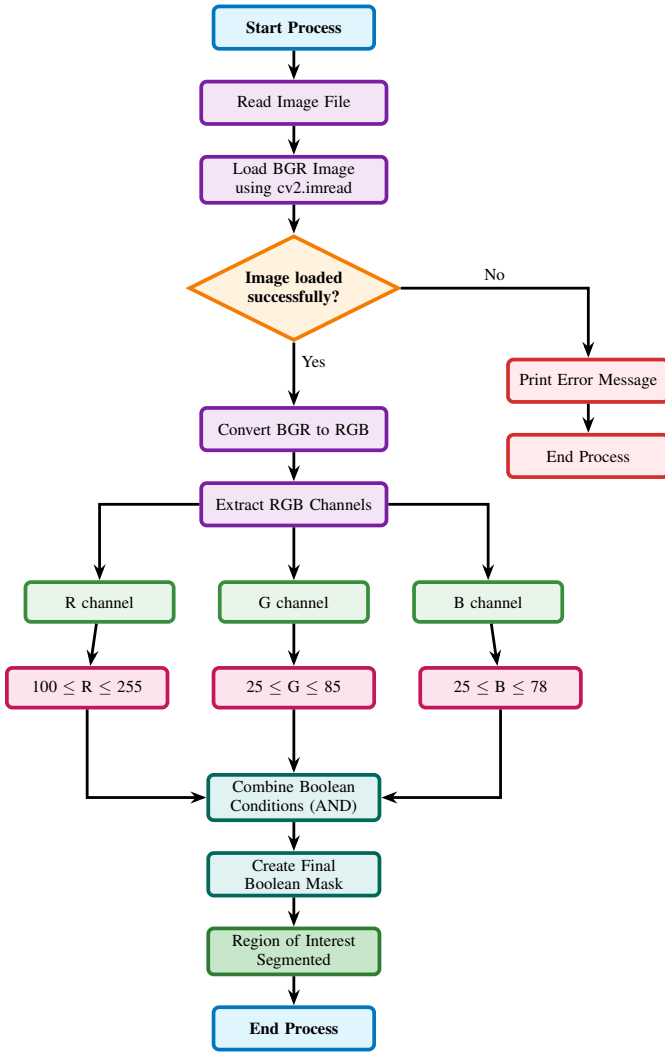


Fig. 9. RGB mask creation flowchart for region of interest segmentation.

PNG, and saved to the database as a Batch object. The process can be appreciated in Fig. 14 and an example is shown in Fig. 13.

D. Dataset Description The dataset was built from images captured during controlled FVH production cycles, covering multiple developmental stages from germination (day 1) through full growth (day 14). A total of 1,669 raw images were collected across both camera positions. From these, 11,688 patches of 250×250 pixels were extracted using the sliding window procedure described above: 7,753 from the superior view (3,730 healthy, 4,023 diseased) and 3,935 from the inferior view (1,978 healthy, 1,957 diseased). The full dataset was split into training (70%), validation (15%), and test (15%) subsets. The dataset composition and split are summarized in Table II.

The *diseased* class encompasses visual anomalies that arise from common problems observed during FVH production cycles, particularly in batches where full control of the technique was not maintained. The specific pathologies and alterations

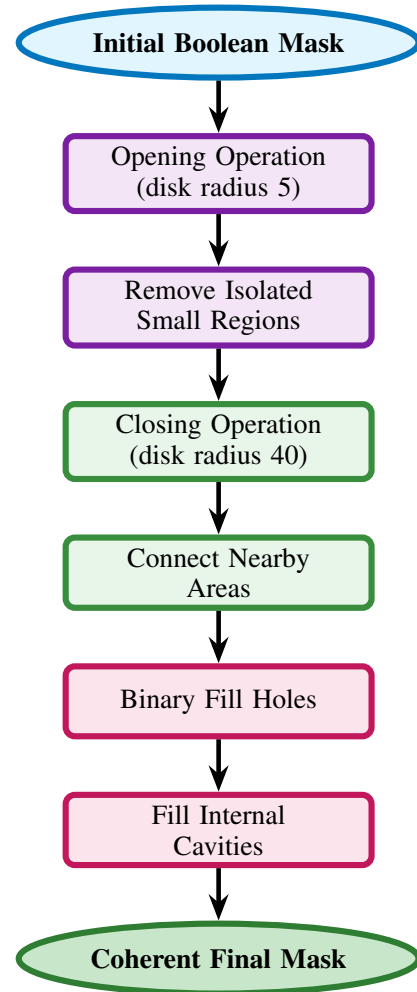


Fig. 10. Sequential morphological operations applied to the boolean mask.

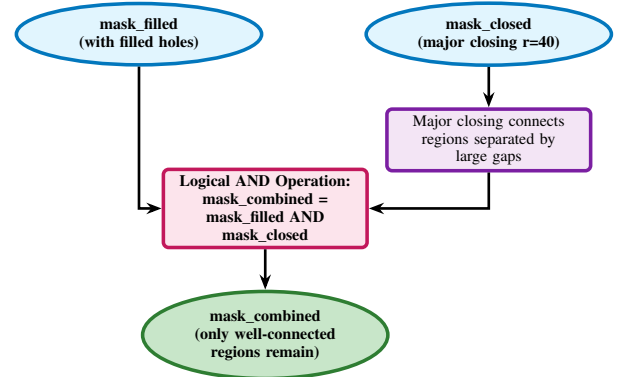


Fig. 11. Mask combination using logical AND operation.

included in this class are: (1) **dark spots**, appearing as localized black areas on the leaf surface, associated with necrotic tissue or contact damage; (2) **color shift in the grain**, visible as discoloration of the maize kernel during germination; (3) **fungal colonization**, identifiable by a cotton-like white or grayish texture spreading across the aerial and root surfaces, promoted by excess humidity or insufficient ventilation; (4)

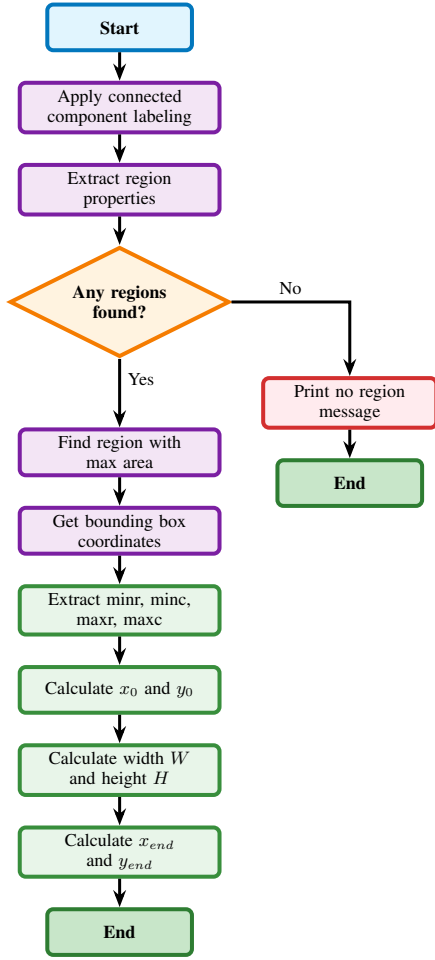


Fig. 12. Bounding box detection flowchart for the region of interest.

TABLE II
Dataset Split by Subset and Class (Combined Views)

Subset	Class	Patches	Subtotal	Total
Train (70%)	Healthy	3,996	8,182	11,688
	Diseased	4,186		
Validation (15%)	Healthy	856	1,753	
	Diseased	897		
Test (15%)	Healthy	856	1,753	
	Diseased	897		

root rot, detectable from the inferior view as a color transition from white to brown or dark tones in the root mass; and (5) **chlorosis and general green discoloration**, characterized by a loss or change of the normal green pigmentation of the leaf blade, typically associated with nutrient imbalances, excess or insufficient irrigation, or inadequate photoperiod conditions.

These anomaly categories were selected based on their prevalence during the experimental cycles conducted at the UPIITA-IPN hydroponic laboratory and their visual distinguishability under the imaging conditions of the monitoring system.

E. Deep Network The classification was based on the CNN MobileNet architecture, with the objective of identifying anomalies in images and assigning them categories of “health”

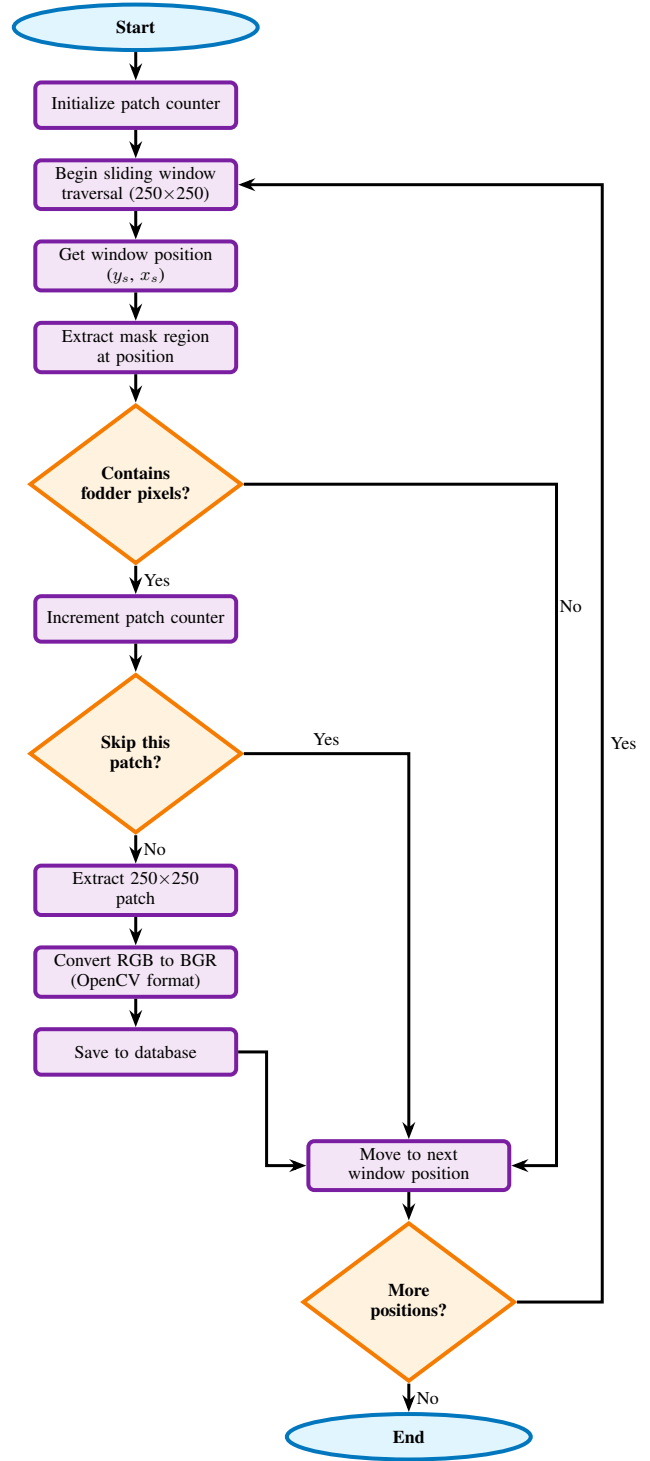


Fig. 13. Flowchart of patch extraction process using a 250×250 pixel sliding window.

or “disease”. MobileNet architectures have demonstrated excellent performance in plant disease detection tasks while maintaining computational efficiency suitable for deployment on resource-constrained devices [7]. The depthwise separable convolutions employed in MobileNetV2 significantly reduce the number of parameters while preserving classification ac-

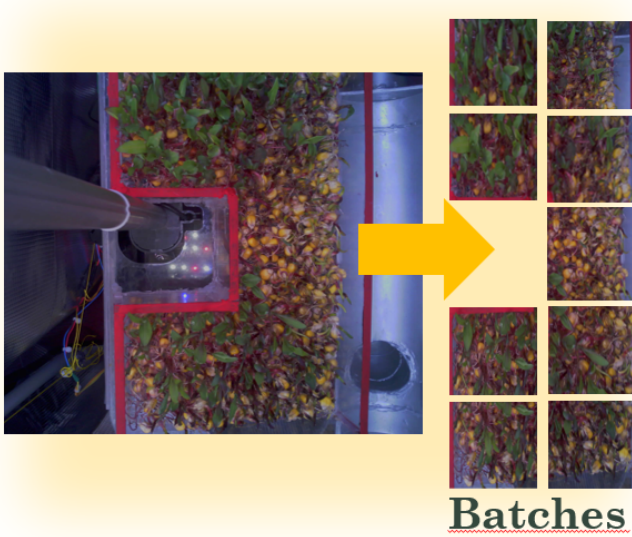


Fig. 14. Batch creation.

curacy, making it particularly suitable for agricultural applications requiring real-time processing [8].

To carry out this task, firstly, a filtering process of the raw data was performed, corresponding to the photographic shots, followed by preprocessing and labelling stages. Additionally, and in order to address the visual variations inherent to each capture angle, the decision was made to train two independent models: one specialised in images obtained from the superior position and another in those corresponding to the inferior position.

1) *MobileNetV2 architecture* MobileNetV2 [13] is a lightweight convolutional neural network designed for mobile and embedded applications. Its architecture is built upon an *inverted residual* structure in which the shortcut connections are placed between the bottleneck layers rather than between the expanded representations, as is the case in standard residual networks. This design substantially reduces the memory footprint required during inference while preserving representational capacity.

The network consists of 19 bottleneck residual blocks organized in a progressive sequence that gradually reduces the spatial feature resolution from 224×224 pixels down to 7×7 pixels. Each bottleneck block applies: (1) a 1×1 pointwise convolution that expands the channel dimensionality by an expansion factor t (typically $t = 6$); (2) a 3×3 depthwise convolution that processes each channel independently; and (3) a 1×1 pointwise projection that reduces the channels back to the bottleneck dimension. The use of ReLU6 as the activation function contributes to numerical robustness on hardware with limited floating-point precision.

In this work, MobileNetV2 was employed with weights pre-trained on the ImageNet dataset (transfer learning). The convolutional base served as a fixed feature extractor, and only the final classification layers were replaced and retrained to adapt the model to the specific visual characteristics of hydroponic

maize fodder. A custom classification head was appended to the base model, consisting of a fully connected layer for 2-class output, a softmax activation layer, and a classification layer. A dropout layer with a rate of 0.60 was included for regularization. The complete architecture is summarized in Table III.

TABLE III
MobileNetV2-Based Classifier Architecture Summary

Component	Output	Trainable
Input layer	$224 \times 224 \times 3$	No
MobileNetV2 base (frozen)	$7 \times 7 \times 1280$	No
Dropout (rate = 0.60)	$7 \times 7 \times 1280$	No
Fully Connected (2 cls.)	2	Yes
Softmax	2	No
Classification layer	2	No

Training was performed using the Adam optimizer with the hyperparameters described in equation form. The cross-entropy loss function used for the binary classification task is defined as:

$$\mathcal{L}_{CE} = -\frac{1}{N} \sum_{i=1}^N [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)] \quad (1)$$

where N denotes the mini-batch size, $y_i \in \{0, 1\}$ is the ground-truth label of the i -th sample (0 for healthy, 1 for diseased) and $\hat{y}_i \in [0, 1]$ is the predicted probability produced by the softmax layer. The training configuration used an initial learning rate of 1×10^{-4} , L_2 regularization of 0.005, a piecewise learning rate schedule with a drop factor of 0.1 every 5 epochs, a mini-batch size of 8, and a maximum of 3 epochs per session with early stopping (patience of 5 validation evaluations without improvement). GPU execution was used to accelerate training.

2) *Image classifier for superior view* All the preprocessed image batches were organized into two directories, corresponding to the “health” and “disease” classes. The data were randomly divided maintaining class balance: 70% for training, 15% for validation, and 15% for testing. This distribution ensures randomness, avoids bias introduction, and guarantees that the validation and test data are completely novel to the network.

Data augmentation was applied exclusively to the training set to increase dataset variability and improve model robustness, as shown in Table IV. The augmentation was implemented using MATLAB’s `augmentedImageDatastore`, which applies transformations on-the-fly during training without generating additional files on disk; each epoch presents the model with differently transformed versions of the same original patches, preserving the fixed sample count while effectively increasing the diversity of representations seen during learning. The augmentation strategy follows established best practices for agricultural image datasets, incorporating geometric transformations that simulate real-world variations in camera positioning and environmental conditions [14]. These intentionally abrupt variations were designed to make the network more robust to different viewing conditions.

TABLE IV
Data Augmentation Strategy Applied to Each Dataset Subset

Subset	Transformations	Augmented
Training	• Resize 224×224 • Rotation (-50° to $+50^\circ$) • X/Y translation ($\pm 30-60$ px) • Horizontal reflection • Scaling ($0.3-1.5 \times$)	Yes
Validation	Resize 224×224	No
Test	Resize 224×224	No

The training was configured with early stopping after 5 consecutive evaluations without improvement in the validation set to avoid overfitting. Four independent training sessions were conducted to calculate performance averages, with results shown in Table V and Table VI.

TABLE V
MobileNetV2 Performance for Healthy and Diseased Classification, Superior View

Metric	S1	S2	S3	S4	Mean	SD
Accuracy	94.41	95.01	94.15	90.45	93.51	1.97
Precision	98.21	98.92	98.37	99.40	98.73	0.50
Sensitivity	90.88	91.38	90.22	82.09	88.64	4.23
F1-Score	94.40	95.00	94.12	89.92	93.36	2.25
Specificity	98.21	98.93	98.39	99.46	98.75	0.56
All values in percentage (%).						

TABLE VI
MobileNetV2 Performance for Healthy and Diseased Classification, Inferior View

Metric	S1	S2	S3	S4	Mean	SD
Accuracy	98.98	96.77	98.47	97.28	97.88	0.99
Precision	99.65	95.36	99.65	95.11	97.44	2.42
Sensitivity	98.29	98.29	97.27	99.66	98.38	0.95
F1-Score	98.97	96.81	98.45	97.33	97.89	0.96
Specificity	99.66	95.27	99.66	94.93	97.38	2.46
All values in percentage (%).						

To select the optimal model for implementation in the online system, global accuracy and sensitivity were evaluated. This criterion prioritizes the correct identification of truly healthy instances, minimizing the risk of classifying diseased plants as healthy. From this analysis, the second training session (S2) for the superior view achieved the highest sensitivity of 91.38%, making it the selected model for deployment.

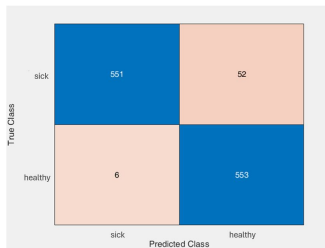


Fig. 15. Confusion matrix, superior shot.

For the “Inferior View”, the fourth model from the training sessions stands out, which although it did not achieve maxi-

mum global accuracy, exhibits the best balance between the parameters considered.

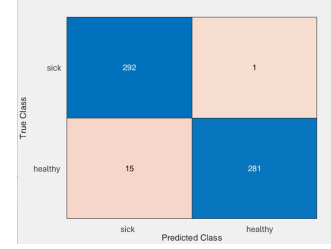


Fig. 16. Confusion matrix, inferior shot.

F. Height Measurement The next step is to estimate the height of the green fodder with respect to the base of the tray. Fig. 18 shows the result of the green fodder height quantification process, as well as the segmentation of the plant area, the automatic identification of the upper point of the green fodder and the tray edge, used for height calculation. This process consists of:

- 1) **Orientation correction:** The image is in horizontal format, it is rotated to be vertical.
- 2) **Dark edge cropping:** Rows and columns with low illumination are automatically eliminated to isolate the relevant crop region.
- 3) **Colour segmentation (HSV):** A threshold is applied to isolate the green and yellow regions that represent the green fodder.
- 4) **Principal component selection:** The object with the largest connected area is identified to define the main green fodder.
- 5) **Red tray detection:** The lower half of the image is analysed searching for red regions. If none is detected, an alternative algorithm is applied using Canny edges and Hough transform.
- 6) **Height calculation:** The distance in pixels between the upper limit of the green fodder and the upper part of the tray is measured. Specifically, the maximum height point is identified by selecting the topmost pixel (y_{top}) from the segmented fodder region across all horizontal positions. This pixel distance is then converted to centimeters using a calibration factor derived from the known physical height of the tray and its corresponding pixel height in the image, following the relationship expressed in (2).

$$h_{cm} = \frac{h_{px} \cdot H_{cm}^{tray}}{H_{px}^{tray}} \quad (2)$$

where h_{cm} is the estimated fodder height in centimeters, h_{px} is the measured fodder height in pixels, H_{cm}^{tray} is the known physical tray height (used as the reference measurement), and H_{px}^{tray} is the tray height in pixels detected in the same image.

The implementation of the height calculation algorithm is presented in Fig. 17.

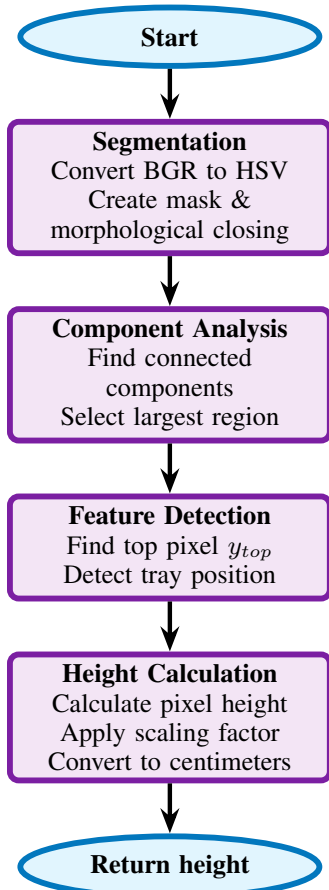


Fig. 17. Height calculation workflow for green fodder measurement.

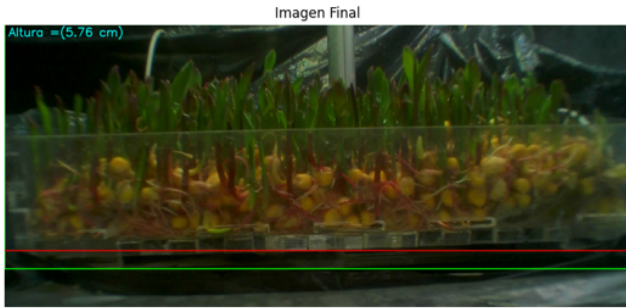


Fig. 18. Final image.

IV. CONCLUSION

This work presents a comprehensive automated monitoring system for hydroponic maize green fodder that integrates classical image processing with efficient deep learning architectures. The main contributions of this research are:

(1) Specialized system for hydroponic environments: Unlike existing works that focus primarily on field crops [5], [6], our system is specifically designed for controlled hydroponic conditions, addressing unique challenges such as water reflections, translucent trays, and non-uniform illumination from single-point light sources.

(2) Edge-compatible deployment: The MobileNetV2-based classification achieves competitive accuracy (95.01% superior view, 97.28% inferior view) comparable to state-of-the-art approaches using ResNet or DenseNet [9], [10], while operating on low-cost ESP32-CAM hardware. This represents a significant advancement over existing systems that require high-end computing infrastructure.

(3) Multi-view integrated assessment: The combination of superior and inferior camera viewpoints provides complementary diagnostic information, enhancing reliability compared to single-view approaches. The integration of simultaneous height measurement enables comprehensive crop monitoring beyond disease detection alone.

(4) Robust preprocessing pipeline: The HSV-based segmentation combined with adaptive image enhancement (CLAHE, gamma correction) [11], [12] effectively handles challenging illumination conditions typical of hydroponic facilities, requiring minimal manual intervention.

Compared to related works (Table I), our system achieves competitive accuracy while offering unique advantages: deployment on resource-constrained hardware, specialization for hydroponic environments, and integrated growth monitoring. **A particularly relevant contribution of this work lies in the patch-based extraction strategy** (250×250 pixels): by sampling multiple sub-regions from a single image of the same crop, the procedure intrinsically captures the morphological and chromatic heterogeneity present across different zones of the tray—variations in plant density, illumination response, root development and emerging anomalies. This treats every captured image as a rich source of diverse training samples rather than a single observation, providing an implicit form of data augmentation **without the need to generate additional images or grow additional crops**. This is a decisive practical advantage in hydroponic fodder research, where the limited number of production cycles, the duration of each cycle (typically 8–14 days), and the cost of running parallel trays are well-known bottlenecks for the validation of computer-vision models. In other words, the same physical experiment yields substantially more usable, statistically diverse data than a whole-image approach would, while the strategic morphological operations ensure robust region-of-interest extraction even under the suboptimal imaging conditions characteristic of hydroponic facilities.

The preprocessing pipeline proved effective in handling challenging illumination, shadows, and reflections inherent to hydroponic systems. The data augmentation strategy [14] successfully improved model robustness to geometric variations, achieving stable performance across multiple training sessions (mean accuracy: $93.51\% \pm 1.97\%$ superior view, $97.88\% \pm 0.99\%$ inferior view).

The developed web-based system integrates all processing stages into a unified platform accessible for real-time monitoring, facilitating early decision-making for crop management. This accessibility is particularly valuable in agricultural contexts where timely intervention can prevent disease spread and

optimize production yields.

Scaling this prototype to a large-scale commercial hydroponic facility would require addressing several technical aspects. On the connectivity side, each sensor node (ESP32-CAM) would need to operate over a robust local network — whether wired Ethernet or industrial Wi-Fi — capable of handling simultaneous image transmissions from multiple trays without packet loss. A lightweight edge gateway, such as a Raspberry Pi or equivalent single-board computer, could aggregate inference results locally and forward only the diagnostic summaries to a central server, reducing bandwidth requirements. Regarding energy autonomy, nodes in facilities without permanent AC power could be powered through low-voltage DC buses shared with LED lighting circuits, or supplemented with small photovoltaic panels where natural light is available. Standardizing the camera mounting fixtures would also be important to preserve the geometric calibration assumptions on which the height measurement algorithm relies, particularly the known tray dimensions used as the pixel-to-centimeter reference.

Limitations and future work: The current system was validated on a limited disease set in controlled laboratory conditions. Future work should: (1) expand the dataset to include additional disease types and environmental variations, (2) conduct field validation in operational hydroponic facilities, (3) explore attention mechanisms to improve feature extraction [8], (4) implement temporal analysis for disease progression tracking, and (5) extend the system to other hydroponic crops beyond maize fodder.

ACKNOWLEDGMENT

The authors acknowledge the support provided by the Secretaría de Ciencia, Humanidades, Tecnología e Innovación (SECIHTI) through a postgraduate scholarship, and the research funding received from the Instituto Politécnico Nacional through grant 2024-A227 under the program "Proyectos de Desarrollo Tecnológico o Innovación para Alumnas y Alumnos del IPN" (PRODTIA).

REFERENCES

- [1] M. S. Ahamed, M. M. Rahman, M. Sultan, M. Aleem, R. R. Shamshiri, and S. K. Balasundram, "Present status and challenges of fodder production in controlled environments: A review," *Smart Agricultural Technology*, vol. 3, p. 100080, 2023.
- [2] A. Kamilaris and F. X. Prenafeta-Boldú, "Deep learning in agriculture: A survey," *Computers and Electronics in Agriculture*, vol. 147, pp. 70–90, 2018.
- [3] B. Chen and S. Chen, "Deep learning," in *Machine Vision Technology*. Singapore: Springer Nature Singapore, 2026, pp. 273–297.
- [4] N. Shelar, S. Shinde, S. Sawant, S. Dhumal, and K. Fakir, "Plant disease detection using cnn," in *ITM Web of Conferences*, vol. 44. EDP Sciences, 2022, p. 03049.
- [5] A. Upadhyay *et al.*, "Deep learning and computer vision in plant disease detection: A comprehensive review of techniques, models, and trends in precision agriculture," *Artificial Intelligence Review*, vol. 58, no. 2, p. Article 29, 2025.
- [6] M. Shoaib, B. Shah, S. El-Sappagh, A. Ali, A. Ullah, F. Alenezi, T. Gechev, T. Hussain, and F. Ali, "An advanced deep learning models-based plant disease detection: A review of recent research," *Frontiers in Plant Science*, vol. 14, p. 1158933, 2023.
- [7] C. Bi, S. Xu, N. Hu, S. Zhang, Z. Zhu, and H. Yu, "Identification method of corn leaf disease based on improved mobilenetv3 model," *Agronomy*, vol. 13, no. 2, p. 300, 2023.
- [8] B. Khomkham and Y. Pankaseam, "Lightweight convolutional neural network model based on modified mobilenetv3 for plant disease classification," *SN Computer Science*, vol. 6, no. 1, p. 61, 2025.
- [9] M. Shoaib, A. Sadeghi-Niaraki, F. Ali, I. Hussain, and S. Khalid, "Leveraging deep learning for plant disease and pest detection: a comprehensive review and future directions," *Frontiers in Plant Science*, vol. 16, p. 1538163, 2025.
- [10] A. Y. Ashurov, M. S. A. M. Al-Gaashani, N. A. Samee, R. Alkanhel, G. Atteia, H. A. Abdallah, and M. S. A. Muthanna, "Enhancing plant disease detection through deep learning: a depthwise cnn with squeeze and excitation integration and residual skip connections," *Frontiers in Plant Science*, vol. 15, p. 1505857, 2024.
- [11] K. Zuiderveld, "Contrast limited adaptive histogram equalization," in *Graphics Gems IV*, P. S. Heckbert, Ed. San Diego, CA, USA: Academic Press Professional, Inc., 1994, pp. 474–485.
- [12] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 4th ed. Upper Saddle River, NJ, USA: Pearson, 2018.
- [13] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4510–4520, 2018.
- [14] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 1, pp. 1–48, 2019.