Neural Computing and Applications
https://doi.org/10.1007/s00521-025-10973-5

ORIGINAL ARTICLE

ViT-SENet-Tom: machine learning-based novel hybrid squeeze–excitation network and vision transformer framework for tomato fruits classification

S M Masfequier Rahman Swapno[1] · S. M. Nuruzzaman Nobel[1] · Md Babul Islam[2] · Pronaya Bhattacharya[3] · Ebrahim A. Mattar[4]

Received: 18 July 2024 / Accepted: 12 December 2024

Abstract

Tomatoes are essential fruits in numerous nations for their vast demand. It is very important to maintain the freshness of tomatoes. One of the primary challenges in the recent culinary landscape is accurately identifying healthy tomatoes while effectively eliminating damaged or rejected ones. Existing approaches employ various strategies for categorizing tomato fruit, but they often suffer from inaccuracies, slow detection, and suboptimal performance. Thus, motivated by this gap, in this paper, we propose a novel machine learning (ML) framework, *ViT-SENet-Tom*, which is a hybrid vision transformer (ViT) model with squeeze and excitation (SENet) block network for fast, accurate, and efficient tomato fruit classification. The framework works on three tomato classes, respectively, the ripe, unripe, and reject. In developing the proposed model, we utilized advanced and newly designed layers and functions. This integration created a more complex and sophisticated neural network, significantly enhancing efficiency and contributing to the model's novelty. Our chosen dataset was small initially, but we implemented augmentation techniques to increase its size. This approach made our system more reliable, efficient, and effective. The hybrid *ViT-SENet* framework employs encoders and self-attention networks with squeeze and excitation channel functions to allow precise, robust, fast, and efficient tomato classification. In simulation, the framework achieves a training accuracy of 99.87% and validation accuracy of 93.87%, indicating the precise classification of tomatoes. Besides, this work tests accuracy using fivefold cross-validation. The highest accuracy seen at fold-5 is 99.90%. These testing results demonstrate the efficacy of the proposed framework in real-deployment scenarios. The implementation has the potential to provide enhanced and more sustainable food security and safety in future.

Keywords Advance neural network · Food safety · Machine learning · Tomato fruit · Fruit classification · Vision transformer · SENet · Agriculture · Fresh fruits

1 Introduction

Tomatoes are an important crop that can be automated appropriately to increase crop yields, preserve productivity, and ensure ongoing production. In many nations, the transformation of tomato farming via smart agricultural techniques and artificial intelligence (AI) is leveraged by robots. Robotic harvesting [1] involves the collection of fruit with a robot arm and detection via a computer vision system. Thus, it is imperative to identify the tomato quality, so robotic arm can pick just the ripe and consumable fruit, leaving the others on the branch or vine to ripen. In the past ten years, many fruit detection methods have been created. Color, texture, form, and other superficial aspects of the picture are the primary elements used by conventional methods for detection.

A critical indicator of ripeness in tomatoes is their color. Tomato fruit experiences five distinct phases of development. Their color differences [2], starting as green and progressing to light pink, pink, light red, and finally crimson, categorize them into five separate groups. Typically, a tomato [3] takes 21–28 days to break, 15–20 days to turn, 7–14 days to change pink, 5–6 days to turn light red, and 2–4 days for red phases once it is green. A fresh tomato needs to be classified from the bad ones by the harvesting robot in operation, to fetch the premium price in

Extended author information available on the last page of the article

Published online: 20 January 2025

market. Other factors coupled with this involve the logistics (distance covered during the shipping process) and storage (time duration at the storage warehouse). These factors make it crucial to enhance the tomato categorization system.

A number of recent studies have focused on intelligent agricultural goods processing and sorting using machine learning and pattern recognition techniques. In particular, machine learning is one of the most crucial [4] components of the harvesting robot. However, skilled staff members choose [5] the machine learning-based techniques. These approaches need more flexibility and timeliness to be implemented in agricultural businesses. Additionally, creating a system with such strong performance in terms of timeliness, accuracy, and scalability ought to address several complex problems.

1.1 Motivation and contribution of this research

Our research is motivated by the potential impact of such technology on food security and safety. Tomatoes are an essential fruit in numerous culinary traditions worldwide, making the quality and freshness of tomatoes a critical issue for farmers, wholesalers, and consumers. Traditional methods of tomato classification are often labor-intensive, subjective, and prone to error. By harnessing the power of machine learning and computer vision, we aim to revolutionize this process, providing a more accurate, consistent, and efficient means of determining tomato maturity levels. Accurately classifying tomatoes into ripe, unripe, or rejected categories enhances the efficiency of food production and distribution and ensures that consumers receive fresh, high-quality produce. Additionally, this research aligns with broader sustainability goals by reducing food waste and optimizing resource utilization in agricultural practices. This novel approach's successful implementation showcases machine learning's transformative capabilities in addressing real-world challenges. Through this research, we aim to pave the way for broader adoption of advanced technologies in agricultural and food industries, ultimately contributing to enhanced food security and sustainable practices in future.

Our automated tomato fruit classification system significantly advances efficiency and accuracy. This study focuses on three distinct categories of tomato fruits: ripe, unripe, and reject. Figure 1 displays the three tomato varieties that were classified in this study. Through the innovative use of our hybrid ViT-SENet approach, we have achieved accurate results in classifying tomato fruits. This system delivers rapid processing and ensures exceptional accuracy, making it a valuable tool for agricultural applications and research. In our approach, we offer a effective solution that sets a new standard in tomato fruit classification.

In short, these are the key contribution of the research as follows:

- Development of a novel *ViT-SENet* model dedicated to effective fruits image classification task. This model presents a precise, fast, and fully automatic approach to classify tomato fruits. Its superiority lies in its advanced capability to perform classification tasks more effectively than other existing methods. The model ensures high accuracy and significantly reduces the time required for fruit classification, making it an invaluable tool in food safety domain.
- A novel and advanced layer-based neural network has been created to construct a powerful model for tomato fruit classification. This advance network outlines increased efficiency, driving the model toward faster and more accurate results.
- Developing an advanced, high-precision tomato fruit system that outperforms existing solutions in both speed and accuracy. This system establishes remarkable detection standards, setting a new benchmark for accuracy and efficiency in the field of food safety and security.

1.2 Organization of the paper

This study is divided into several sections. In Sect. 2, you will find an overview of the previous research. Section 3 describes the research plan and methodology used in the study. Section 4 presents the study's findings and analyses. Section 5 offers a comprehensive analysis of the subject matter and a meticulous assessment of the findings and their consequences. Finally, in Sect. 6, the final analysis outlines the research's conclusions and future directions.

2 Related works

In the past ten years, numerous AI investigations have been carried out to identify healthy and fresh food by automated way. Numerous datasets are available for the scientific community. Several works have been carried out utilizing machine learning [6–9], deep learning [10–14] and reinforcement learning [15–18] techniques to build automated system. These approaches outline the automatic, fast and reliable tomato fruit classification, significantly improving

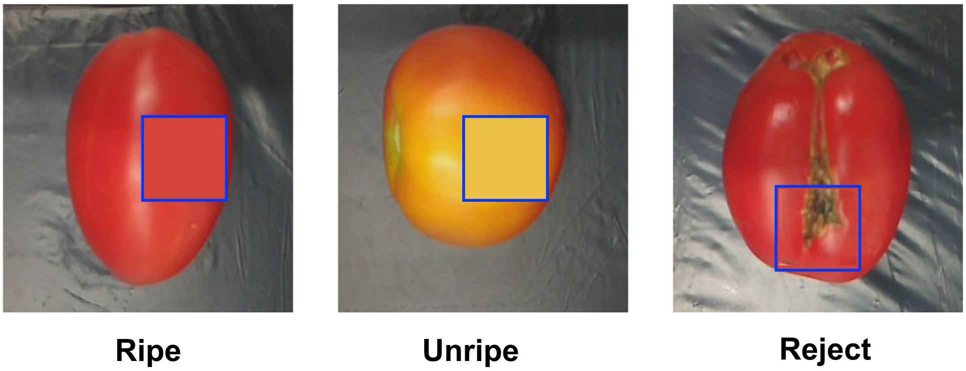


Fig. 1 Sample Images for Tomato Classification into Ripe, Unripe, and Reject Categories

food security results. Every technique has specific constraints and breakthroughs within the medical field that aid healthcare professionals in decision-making. Oliveira et al. [19] utilized data from mass spectrometry. They used direct-infusion electrospray-ionization mass spectrometry and silica gel plates to analyze tomato samples. The writers put into practice a decision tree technique designed for data analysis. The model developed by the authors proved to be 92% accurate, 94% sensitive, and 90% precise in classifying the fruits. Tao et al. [20] proposed a generic intelligent tomato classification system. The authors created a functional tomato classification system with transfer learning and DenseNet-201 that achieved good accuracy in various picture quality, even with high noise levels. In just 29 milliseconds, their algorithm successfully categorized a single tomato picture, demonstrating its enormous potential for practical use. Final accuracy was impacted by the model's performance, which was determined by its recognition regions during categorization. Mahmoud et al. [21] investigated and demonstrated how vital tomato juice is for people with diabetes to lower their platelet activity, preventing them from forming potentially fatal blood clots. Using a data set comprising around 5,266 images of seven different tomato species, they developed a tomato categorization method. The deep learning method, neural network (NN) algorithms, frequently used in picture recognition, was employed for this assignment. Pereira et al. [22] presented a method that uses random forests and digital photography to forecast when papaya fruit will ripen. They achieved 94.30% accuracy for classification when the color features are computed from the peel, which has a minimal computational cost. Chakraborty [23] proposed CNN-based model for obtaining features from an image and a MobileNetV2 model that employs Max Pooling and Average Pooling to identify rotten fruits. Using Max Pooling and Average Pooling, the model achieved an accuracy of 94.97% and 93.72%, respectively, on a dataset comprising three different fruit varieties. Worasawate et al. [24] developed four classifiers for the classification of mango ripeness stages during harvest using machine learning. The authors used four classifiers such as naive-Bayes, feed forward neural network, support vector machine, and k-means. The classifiers were tested on physical characteristics like weight and skin color, as well as electrical qualities like capacitance and voltage, after being trained on the biochemical attributes of mangos such as titratable acidity and total soluble solids. In comparison with other classifiers, the four classifier performed well, with an accuracy of 89.6%.

Fei et al. [25] applied deep learning (DL) technique with Nature Greenhouse for tomato maturity categorization. The SE-YOLOv3-MobileNetV1 model was implemented to classify tomato ripeness. They focused on model speed and accuracy for tomato maturity categorization. Against confirm detection performance, they compare their model's accuracy and fast working than YOLOv3 method, YOLOv3-MobileNetV1, and YOLOv5 models. Authors proposed model achieved an average accuracy of 97.5%. Comparing their recommended model to YOLOv3 and YOLOv5, the latter showed detection speeds of 278.6 and 236.8 ms. Using image processing algorithms, Laykin et al. [26] worked on tomato color, consistency, defects, shape, and stem detection that were tested. Authors camera was pointed upwards in the bottom vision cell, while two were angled at sixty degrees toward the fruit in the top vision cell. Fruit stem and form were determined by the lower vision cell, while color, flaws, and homogeneity were evaluated by the higher vision cell. The author's test the system and achieved 90% accuracy of rejected class, 2% were significantly misclassified, 90% of colors were homogeneously classified, 92% were correctly detected, and 100% of stems were recognized. Phan et al. [27] used YOLOv5 and convolutional neural network (CNN) models to classify tomato fruits. The authors classified tomato fruit of two category as ripe and immature. Combining Yolo5m and ResNet-101 yields 92% accurate predictions for ripe and immature tomatoes. They achieved this results by using 4500 image dataset and 200 epoch training methods with a batch size of 128 and a 224 × 224 pixel image size. In comparison, YOLOv5 and the Efficient-B0 model accurately predict damaged tomatoes with 94% accuracy. On one hand, ResNet-101 has an accuracy of 97%; on the other hand, EfficientNet-B0, Yolov5m, and ResNet-50 each have an accuracy of 98%.

The accurate classification of tomato fruits poses a considerable challenge due to the limitations of existing methods regarding precision and reliability. These methods often struggle with distinguishing between various types of tomatoes, such as ripe, unripe, damaged, or diseased fruits. To address these issues, we introduce a novel machine-learning framework, the *ViT-SENet* model. This approach leverages advanced feature extraction and representation capabilities to classify tomato fruits accurately across multiple categories. Our model achieves exceptional accuracy and demonstrates reliable and effective performance in differentiating between various types of tomato fruits, setting a new benchmark in this domain.

3 *ViT-SENet-Tom*: the proposed framework

This study presents the development of a robust classification model based on Hybrid *ViT-SENet* model. We followed several steps to conduct our research. First, we focused on the dataset and augmented it. Then, we used preprocessing techniques to ensure the effective use of the dataset with the model. After that, we experimented with a set of models to check the effectiveness of our system, and we proposed our method. Subsequently, we found some results and conducted further experiments to validate our system. In Fig. 2, we show the work flow diagram that we followed during our research.

3.1 Dataset analysis and preprocessing stage

Our system's construction involved utilizing a Kaggle dataset [28] that consisted of 2400 tomato fruit images. Each class, namely ripe, unripe, and reject, contained 800 images of tomato fruits. The acquisition of this dataset is of utmost importance in training our model to categorize tomatoes accurately according to their visual attributes. In order to optimize the dataset's efficacy, we implemented sophisticated preprocessing procedures to enrich and broaden the image data further. By undergoing this augmentation process, our system enhances its capacity to acquire more resilient characteristics, increased data, and patterns, enhancing its accuracy in classifying tomato fruits.

To preprocess our data, we employ a series of sophisticated preprocessing techniques. We utilized the data transformation process, which involves converting raw data into a suitable format for analysis or modeling purposes. These jobs entails rescaling data, cropping and resizing images, standardizing it, transforming categorical variables into numerical format, handling missing values, and generating new features to improve the quality and suitability of data for machine learning algorithms. Normalization is further done to adjust numerical data to fit within a standardized range, typically from 0 to 1 or -1 to 1. This approach helps to stabilize and speed up the training of machine learning algorithms by ensuring that all characteristics contribute equally to model fitting. It mitigates the dominance of variables with larger scales over the model. This technique improves the effectiveness and utilization of our dataset for modeling purposes. The graphical depiction of the preprocessing operation is illustrated in Fig. 3.

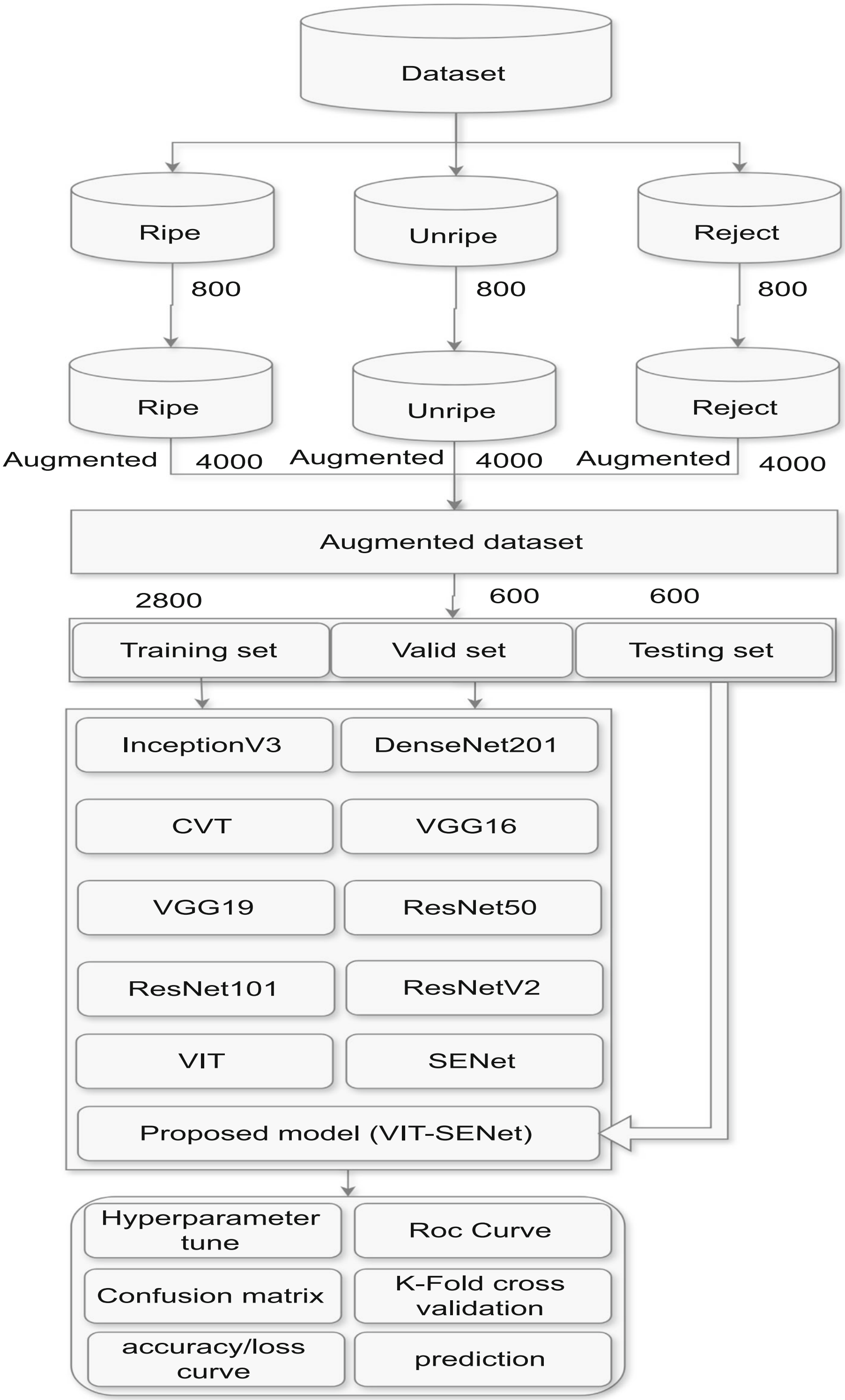
To boost the quantity of images in our dataset, we used an augmentation strategy for our analysis. After implementing this method, the data counts for each of the three classes in our system significantly increased. Table 1 displays the data before augmentation, subsequent to augmentation, our system's training, validation, and testing data, categorized by class. To enhance the accessibility of this approach, we partitioned the complete dataset into three distinct subsets—70% for training, 15% for validation, and an additional 15% for testing.

3.2 The proposed *ViT-SENet* framework

ViT employs encoders, self-attention networks, categorization layers, and multi-linear perceptrons to classify tomatoes accurately. The ViT model utilizes tensor-anchored counts to encode picture and class token embeddings. Incorporating class token embedding vectors from the fine-tuned model leads to an enhancement in classification accuracy. Figure 4 illustrates the implementation of the ViT model.

The self-attention mechanism connects the different components of a sequence, as described in the work by Vaswani [29]. The structure is encoded by the self-attention network and the multi-layer perceptron (MLP) block using a normalizing layer including residual connections. The outcome is obtained by merging keys, value pairs, and

Fig. 2 Presenting the work flow diagram, carried out this research by following these procedures



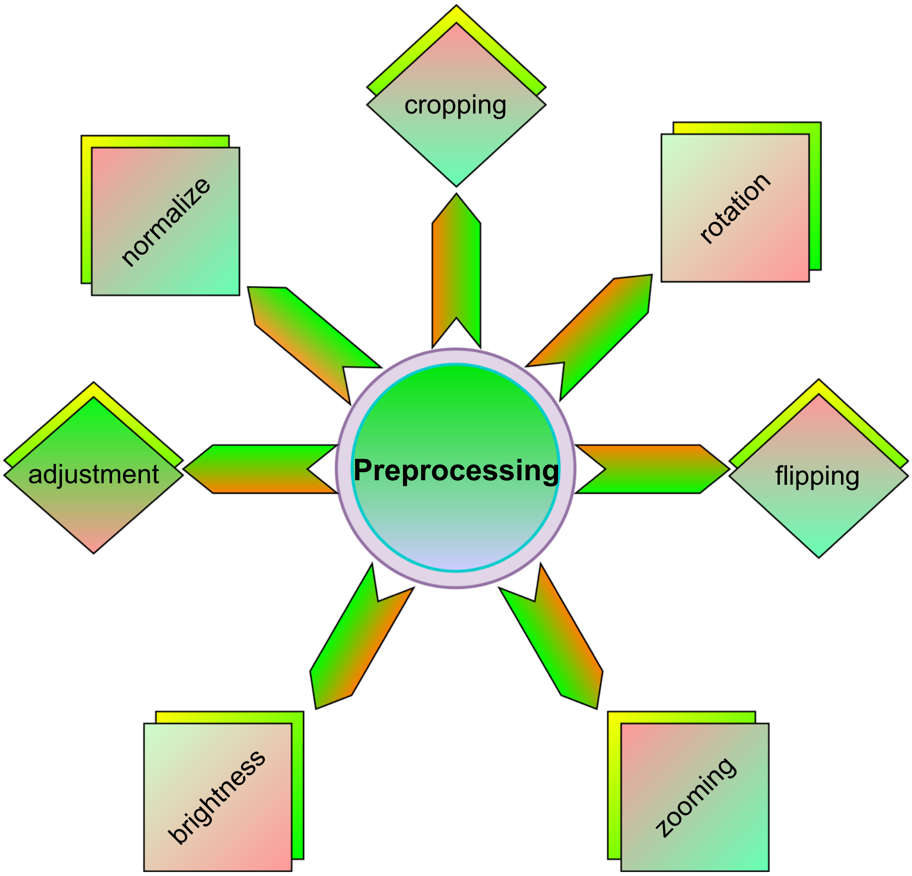


Fig. 3 A set of preprocessing technique that we implemented on dataset for effective use in this system

Table 1 Data distribution before and after augmentation, and splits for training, validation, and testing

Class	Before augment	After augment	Train	Validation	Test
Ripe	800	4000	2800	600	600
Unripe	800	4000	2800	600	600
Reject	800	4000	2800	600	600

searches with meticulous deliberation [30]. The query compatibility function assigns weights to each item and computes the weighted aggregate.

To find the dot product execution, divide each by the square root of d_k, while considering the input dimensions of queries and keys and d_v. The softmax function is utilized to allocate weights to the value pairs. The attention matrix consists of the set of queries (Q), keys (K), and values (V) that are utilized to calculate the attention function simultaneously. The attention (A) calculation (Q, K, V) is executed as follows.

$$A(Q, K, V) = softmax\left(\frac{Q * K^T}{\sqrt{d_k}}\right) * V \tag{1}$$

Through the use of multi-headed attention, the model is able to process information from multiple representation subspaces at once.

$$\begin{aligned}\text{MultiHead}(Q, K, V) \\ = \text{Concat}(\text{head}_1, \ldots, \text{head}_h) * \omega_0\end{aligned} \tag{2}$$

$$\text{head}_i = \text{Attention}\left(QW_i^Q, KW_i^K, VW_i^V\right) \tag{3}$$

Dense and dropout layers make up a multi-layer perforated (MLP) feed-forward neural network. These MLP blocks have identical structures and are stacked with layer blocks. Let us use the token quality X as an example, with a sequence length of n and dimension d. The mathematical definition of each block is presented below:

$$Z = \sigma(XU), \tilde{Z} = s(Z), Y = \tilde{Z}V, \tag{4}$$

$$\begin{aligned}z_0 = \left[x_{\text{class}}; x_p^1 E; x_p^2 E, \ldots, x_p^n E\right] \\ + E_{pos}, E \epsilon R^{(N+1)*D}\end{aligned} \tag{5}$$

$$z_l^1 = MSA(LNz_{l-1}) + z_{l-1}, l = 1, \ldots, L \tag{6}$$

$$z_l = MLP\left(LNz'^l l\right), z_1^l, l = 1, \ldots, L \tag{7}$$

$$y = LN\left(z_l^0\right) \tag{8}$$

The symbol σ represents the activation function. The softmax function is denoted by $s(.)$. The linear projection

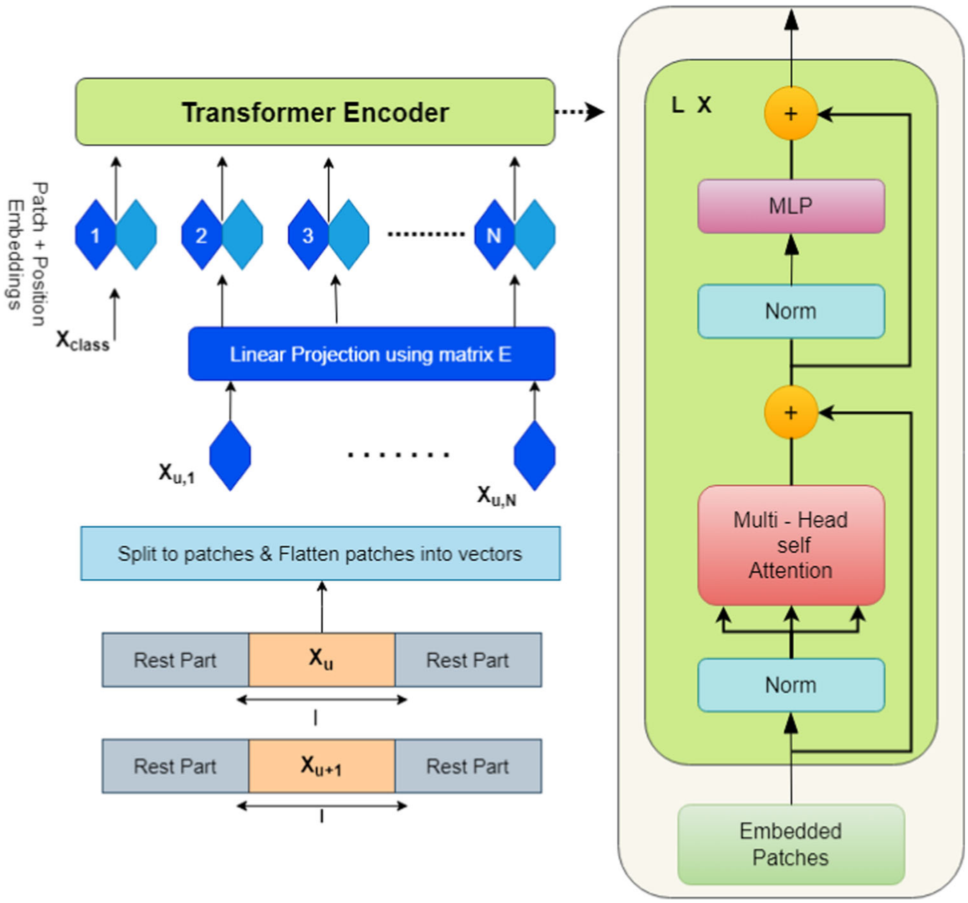


Fig. 4 The architecture of the Vision Transformer model, represent multiple transformer encoder, embedded patches through, using self-attention mechanisms to record contextual information and spatial correlations

dimensions of the channel are represented by the variables U and V. The layer records spatial interaction based on Eq. 4, with individual tokens being computed individually. Prior to being arranged in Eq. 4, the characteristics of the class token, patch embedding, and learnable embedding position of the layers are thoroughly explained in Eqs. 5 and 6. Equation 8 represents the ultimate result produced by the encoder.

$$S(Z) = \text{softmax}\, f(Z) \tag{9}$$

The specifics of the attention block serve as the foundation of our model. This crucial component effectively combines critical aspects of information processing, establishing the foundation for the collaborative functions of the multihead and head characteristics. An essential aspect of our model's architecture is where the MLP is incorporated. The intricate dynamics at work are captured by the MLP's functionality, which enhances the model's capacity to recognize complex patterns and correlations in the data. The seamless integration of these components leads us to a pivotal juncture: utilizing the softmax function. The final stage of our ViT Model consolidates the model's gained features and knowledge, ensuring optimal categorization. Our approach involves meticulous examination of the dataset, skillful preprocessing techniques, and a thoughtfully crafted model architecture.

3.3 The SENet integration

SENet is a channel attention approach, proposed by Huang [31]. To selectively obtain key feature information, the CNN in SENet constantly adjusts the relationships between

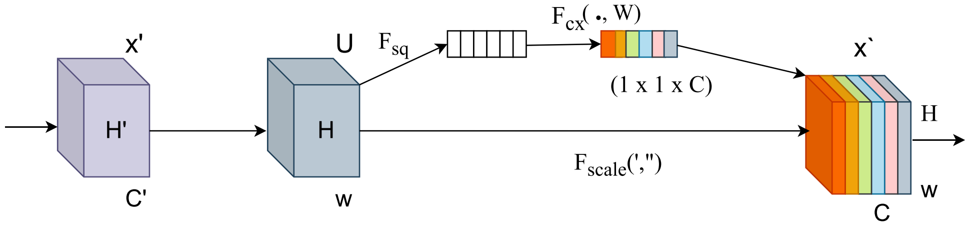


Fig. 5 The SENet's architecture, emphasizing the network's special attention mechanism that improves feature representation in CNN

various feature channels and analyzes and understands their interactions. The diagnostic model can enhance its diagnostic representation and classification capabilities by dynamically altering the importance of each channel and suppressing redundant channels. Figure 5 illustrates the core structure of the SENet, which consists mainly of two essential operations, namely excitation and squeezing operations [32].

The mapping transformation F_{tr}, which uses the excitation function $X \in R^{H' \times W' \times C'}$, $U \in R^{H \times W \times C}$, is used to create matrix U from matrix X, as shown in Fig. 5. Matrix X has a height of $H' \times W' \times C'$, and matrix U has a width of $H \times W \times C$. One way to express the mathematical operations is as follows.

$$u_i = v_i \times X = \sum_{j=1}^{C'} v_i^j \times x_i^j \tag{10}$$

where v_i represents the i^{th} convolutional kernel, u_i denotes the i^{th} sub-matrix in the matrix, v_i^j denotes the j^{th} input in the i^{th} convolutional kernel, and x_i^j denotes the j^{th} input as well.

Matrix U is subjected to global average pooling during the squeeze process, which reduces the spatial dimensions of the feature mappings in U, particularly the $H \times W$ dimensions. A one-of-a-kind value representing the integration of global data is the end product. The result is a feature vector with dimensions $1 \times 1 \times C$, which is the result of averaging all of the channels. Mathematical operations can be symbolically represented.

$$Z_i = F_{\text{sq}}(u_i) = \frac{1}{H \times W} \sum_{m=1}^{H} \sum_{n=1}^{W} u_i(m, n) \tag{11}$$

In this context, Z_i denotes the compressed feature of channel i, and $u_i(m, n)$ denotes the mapped value in the i^{th} channel from the m^{th} row and n^{th} column.

The main goal of the excitation operation is to understand the nonlinear relationships among the feature channels that the squeeze operation outputs. In addition, the weights of each channel are dynamically adjusted to prioritize crucial feature information. Evoking excitement is a straightforward and interdependent procedure. In order to declare the following, the internal components carry out precise mathematical calculations.

$$s_i = F_{\text{ex}}(Z, W) = \sigma(g(Z, W)) = \sigma(W_2 \delta(W_1, Z)) \tag{12}$$

$$W_1 \in R^{\frac{c}{q} \times C}, W_2 \in R^{C \times \frac{c}{q}} \tag{13}$$

In that case, σ represents the ReLU activation function that was used following the dimension reduction. δ is the Sigmoid activation function that is applied once the dimension is expanded. The parameters for the two fully linked layers are denoted as W_1 and W_2. The scaling factor is denoted by q, and the output of channel i after the excitation operation is represented by s_i.

After the excitation procedure, a weight vector is produced with dimensions $1 \times 1 \times C$, where C is the number of channels. This weight vector incorporates the significant channel weights. The last step, the scale operation, involves multiplying the channel weight vector by each element in the original matrix U. The symbolic representation of the mathematical operation is as follows.

$$\tilde{X} = F_{\text{scale}}(u_i, s_i) = s_i \times u_i \tag{14}$$

where the final output feature map of the SENet network is represented by $\tilde{X}$. In the implementation above, our hybrid *ViT-SENet* strategy is functioning effectively. Figure 6 displays the architecture of the hybrid *ViT-SENet-Tom* framework. After the execution, we determine the categorization displayed in this architecture.

The *ViT-SENet* approach commences by partitioning an input picture I with dimensions $H \times W \times C$ into patches and subsequently linearly embedding them. The addition of positional embeddings then follows this. Subsequently, these embeddings undergo a sequence of transformer encoder layers for processing.

Inside each layer, the input undergoes normalization. It is then subjected to a multi-head self-attention process, which allows for identifying both global and local dependencies inside the patches. The attention output is merged with the input using residual connections and then passed through a feedforward neural network, again using residual connections. After processing all layers, the output Z_L of the last layer is subjected to a global average pooling operation in order to derive a descriptor z for every channel. The descriptor z undergoes additional processing using fully connected layers and a sigmoid activation function to provide a scaling vector s.

The output of the last transformer layer, denoted as Z_L, is subsequently scaled per channel by the factor s using element-wise multiplication. After scaling the feature maps, they are pooled globally to provide a representation for classification. This is then followed by a linear transformation to obtain logits. A softmax operation is utilized to calculate class probabilities, which serve as the algorithm's projected outputs. This methodology synergizes the capabilities of Transformer-based vision models such as ViT with the channel-wise augmentation provided by SENet, leading to very effective image classification abilities. Algorithm 1 presents the details of initialization, transformer encoder, SENet integration, and the classification head specifics.

Algorithm 1 *ViT-SENet* Algorithm

```
1. Input: Image $I$ of size $H \times W \times C$
2. Output: Class predictions
3. Initialization:
  (a) Extract patches $X$ from $I$ with patch size $p \times p$
  (b) Linearly embed patches: $X_{\text{embed}} = \text{LinearEmbed}(X)$
  (c) Add positional embeddings to $X_{\text{embed}}$
4. Transformer Encoder:
  (a) Initialize learnable parameters $\theta$
  (b) $Z_0 = X_{\text{embed}}$
  (c) For $l = 1$ to $L$                                  ▷ Loop over transformer layers
      (i) $Z_{l-1} = \text{LayerNorm}(Z_{l-1})$
     (ii) $Z_l = \text{MultiHeadAttention}(Z_{l-1}, Z_{l-1}, Z_{l-1}; \theta)$
    (iii) $Z_l = Z_l + Z_{l-1}$                            ▷ Residual connection
     (iv) $Z_l = \text{LayerNorm}(Z_l)$
      (v) $Z_l = \text{FeedForward}(Z_l; \theta)$          ▷ Feed-forward network
     (vi) $Z_l = Z_l + Z_{l-1}$                            ▷ Residual connection
5. SENet Integration:
  (a) Compute global average pooling over $Z_L$ to get channel-wise descriptor $z$
  (b) Project $z$ using fully connected layers:
      (i) $z' = \text{ReLU}(\text{FC}(z, \frac{C}{r}))$
     (ii) $s = \text{Sigmoid}(\text{FC}(z', C))$
  (c) For $i = 1$ to $H$
      (i) For $j = 1$ to $W$

            (A) $Z_L[i, j, :] = Z_L[i, j, :] \odot s$      ▷ Channel-wise scaling

6. Classification Head:
  (a) Apply global average pooling to $Z_L$ to get $C$ feature vectors
  (b) Linearly transform feature vectors to logits: $Y = \text{Linear}(Z_L)$
  (c) Apply softmax to obtain class probabilities: $\hat{y} = \text{softmax}(Y)$
7. Return: Predicted class probabilities $\hat{y}$
```

3.4 Loss calculation of ViT-SENet model

In machine learning models, the loss function frequently consists of multiple components customized for the particular task. Cross-entropy loss is frequently used in classification tasks to assess the discrepancy between the true and predicted labels, guaranteeing that the model's predictions closely match the real categories. Mean squared error is a quantitative tool used in continuous output situations to assess the correctness of a model by measuring the divergence between the actual and predicted values in regression jobs. Furthermore, Regularization Loss is included to reduce overfitting by penalizing overly intricate models, encouraging generalization, and improving the model's performance on unknown data. When combined, these elements form a robust framework that maximizes the model's generalizability and accuracy.

In the context of cross-entropy loss, a batch of size N represents the number of classes, $y_{i,c}$ denotes the ground-truth label (one-hot encoded) for the i_{th} sample and c_{th} class, and $\hat{y}_{i,c}$ represents the projected probability for the same. The cross-entropy loss can be defined as:

$$L_{CE} = -\frac{1}{N}\sum_{c=1}^{C} y_{i,c}\, log\hat{y}_{i,c} \tag{15}$$

L2 regularization is added. The model parameters are θ_j, and the regularization parameter is λ.

$$L_{\text{reg}} = \lambda \sum_{j} \|\theta_j\|_2^2 \tag{16}$$

The ViT-SENet model's total loss function has the following expression:

$$L_{\text{total}} = L_{CE} + L_{\text{reg}} \tag{17}$$

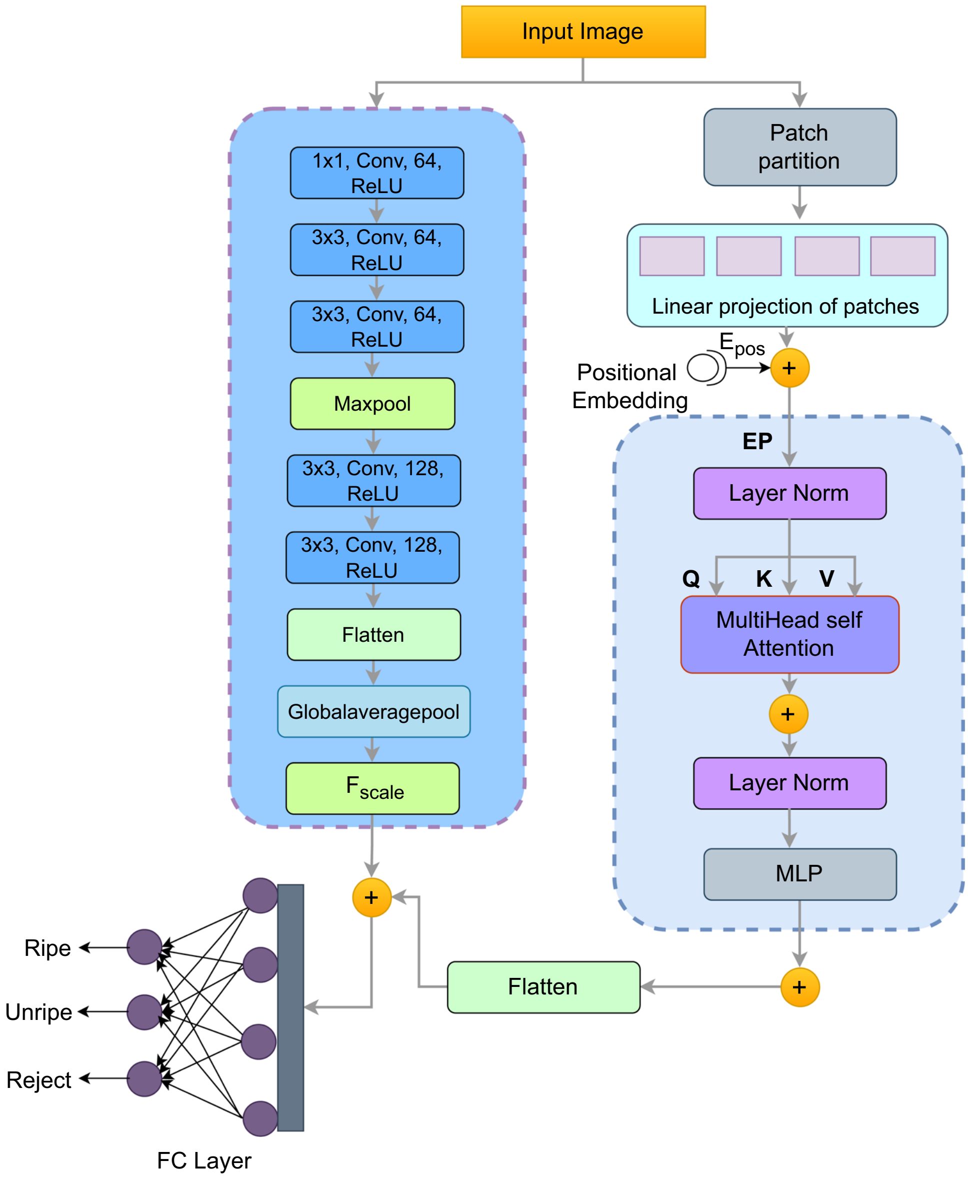


Fig. 6 The proposed *ViT-SENet* integration framework, for enhanced feature extraction and representation in machine learning tasks

A regularization term and the cross-entropy loss are combined to calculate the loss for a ViT-SENet model. The formulation incorporates the unique characteristics of the ViT and SENet components while capturing the typical loss structure for a classification task.

3.5 Hyperparameter tuning of the *ViT-SENet* model

We have extensively collected and structured the hyperparameter tuning techniques for our novel hybrid *ViT-SENet* approach. We thoroughly analyzed several parameters like learning rate, dropout, padding, optimizer, decay, batch size, epoch, and strides. The results of the hyperparameter tuning process for our hybrid *ViT-SENet* model are shown in Table 2.

ViT, initially designed for image classification tasks, utilizes a self-attention mechanism to capture distant relationships within images by employing a transformer-based structure. SENet, in contrast, prioritizes channel-wise feature recalibration to improve the representation of learned features by adaptive recalibration of channel-wise feature responses. Integrating these two architectures into a hybrid model, such as *ViT-SENet*, offers numerous benefits as follows.

- Firstly, it exploits the capability of ViT to capture the whole context and interconnections within the image, enhancing the model's comprehension of intricate linkages within the visual data.
- Furthermore, using SENet's channel-wise recalibration improves features' representation and adaptability,

increasing the model's resilience and efficacy in capturing intricate nuances and fluctuations contained in the data. The model is able to accomplish state-of-the-art performance in computer vision tasks as semantic segmentation, object identification, and picture classification thanks to its hybrid methodology.

- Lastly, the *ViT-SENet* model distinguishes itself by employing a comprehensive strategy for extracting and representing features. It combines ViT's ability to perceive spatial relationships and comprehend global context with SENet's capability to recalibrate features adaptively. This combination effectively tackles significant obstacles in image comprehension, such as managing various item sizes, intricate backgrounds, and lighting and visual characteristics fluctuations.

4 Implementation and results

The suggested *ViT-SENet-Tom* framework's performance analysis is covered in this section. We analyze the performance of tomato fruit classification based on precision, the receiver operating characteristic (ROC) graphs for both image scales generated by our hybrid *ViT-SENet* model. We analyze the loss function of the model. In addition, we incorporate accuracy, recall, F1 score, support metrics, and confusion matrices. The results generated by the utilized models are incorporated into our findings.

4.1 Overview of metrics evaluation and benefits

Performance measures such as training accuracy, validation accuracy, loss, precision, recall, and F1 score are crucial in machine learning for evaluating the effectiveness of a model. Training accuracy quantifies the accuracy of predictions made on the training dataset, whereas validation accuracy evaluates the performance on new and unknown data to identify the presence of overfitting. Loss functions measure the discrepancy between expected and actual values, guiding model optimization. Precision is a metric that quantifies the ratio of accurate optimistic forecasts to all positive predictions, with a specific emphasis on the correctness of the predictions. Recall is a metric that measures the percentage of positives that are correctly identified, focusing on how well the identification is done. The F1 score provides a balanced evaluation of the overall performance of a model by taking into account both precision and recall, using the harmonic mean. These indicators jointly aid in comprehending and enhancing the model's capacity to generalize and produce precise predictions.

4.2 Results showcase from proposed model evaluation

The hybrid *ViT-SENet* model attains a remarkable training accuracy of 99.87% and an excellent validation accuracy of 93.87%. This accomplishment was achieved through rigorous training throughout 50 epochs. The presented results show the model is not overfitting. Figure 7 depicts the training and validation accuracy obtained from the hybrid *ViT-SENet* model. Evaluating machine learning models requires assessing training and validation accuracy since they offer valuable insights into the model's performance and capacity to generalize. Training accuracy quantifies the degree to which the model aligns with the training data, reflecting its learning ability. Validation accuracy, evaluated on unseen data, aids in identifying overfitting and guarantees the model's performance on novel, real-world data.

Our model also has a validation loss of 0.03 and a training loss of 0.001. The training loss measures the degree of fit between the model and the training data, whereas the validation loss assesses the model's performance using untrained data. Monitoring enables the detection of both overfitting and underfitting. A consistently more minor validation loss indicates strong generalization, whereas a higher validation loss than the training loss shows overfitting. This evaluation guides making model improvements to enhance performance. Figure 8

Table 2 Hyperparameter tuning of variables in the hybrid *ViT-SENet* model for optimizing measurement

Parameter	Search space	Selected value
Stride	[3 × 3, 2 × 2]	3 × 3
Optimizer	[Adam, RMSprop, Nadam, Adamax]	Adam
Decay	[0.001, 0.002, 0.0002, 0.00002]	0.0002
Learning Rate	[$1e^{-3}$, $1e^{-4}$, $1e^{-5}$, $2e^{-4}$, $2e^{-5}$]	$2e^{-4}$
Dropout	[0.1]	0.1
Batch Size	[4, 8, 16, 32]	16
Epoch	[40, 50, 55]	50
Padding	[same, valid]	valid

shows the loss that the hybrid *ViT-SENet* model produces during training and validation.

When assessing machine learning models, precision, recall, and F1 score are essential, mainly when dealing with unbalanced datasets. Recall evaluates the model's capacity to find all pertinent occurrences, whereas precision measures the accuracy of optimistic predictions. When recall and precision are equally significant, the F1 score provides a fair evaluation by combining the two metrics. These measures indicate a model's strengths and limitations in managing true positives and false positives, providing a thorough picture of the model's performance beyond mere accuracy. This facilitates choosing and fine-tuning models for the best practical use. Table 3 presents the evaluation matrix in terms of three tomato classes—ripe, unripe, and the reject class.

We conducted a thorough investigation of different ML and DL models to enhance our performance. We assessed different models by considering their training and validation accuracies during this process. Based on our research, we have proposed a hybrid technique, which has shown remarkable efficacy in classifying tomato fruits. We deployed and tested models, including InceptionV3, DenseNet 201, CVT, VGG19, ResNet50, ResNet101, XGBoost, LGBM, ViT, SENet, and our hybrid *ViT-SENet* model. Training accuracy was 98.29%, and validation accuracy was 93.87% for our hybrid *ViT-SENet* model, which expresses exceptional performance, surpassing the performance of all examined techniques. The findings are depicted in Table 4 of our implementation.

To analyze whether overfitting or underfitting occurred, we employed the *k*-fold cross-validation. We used a fivefold cross-validation method in this article. To test how well and how resilient a machine learning model is, fivefold cross-validation divides the data into five equal halves. This approach aids in diminishing the variability of the model assessment and offers a more dependable estimation of its generalization capability compared to a solitary train-test division. In addition, it improves productivity while working with smaller datasets by optimizing data use for testing and training. Testing accuracy peaks at 99.90% during fold-5. Table 5 displays the accuracy for training, validation, and testing using fivefold cross-validation.

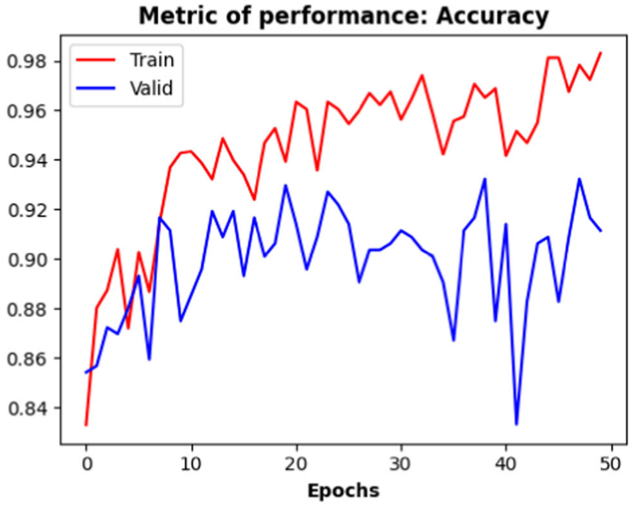


Fig. 7 The accuracy curve in this graph illustrates how well the model performed throughout several epochs or iterations

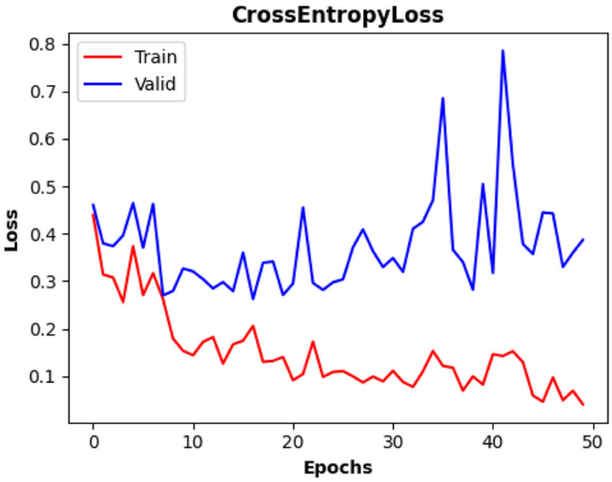


Fig. 8 An illustration that shows the loss curve which present the model efficiency

We also presented the confusion matrix for three classes—ripe, unripe, and reject. The forecast accuracies are remarkable: 95% for ripe, 99% for unripe, and 97% for reject. Figure 9 visually represents the confusion matrix. A confusion matrix is a crucial tool for evaluating the performance of a machine learning model. It comprehensively assesses the model's performance by displaying the number of true positives, true negatives, false positives, and false negatives. This enables the computation of significant metrics such as accuracy, precision, recall, and F1 score. It aids in comprehending the overall precision and the specific categories of mistakes the model produces. Obtaining this detailed information is essential for enhancing the model. Furthermore, it facilitates comparing

Table 3 Evaluation measurement for categorizing three different tomato varieties is shown in this table. For every tomato class, the metrics include support, F1-score, precision, and recall

Class	Precision	Recall	F1 score
Ripe	0.98	0.97	0.97
Unripe	0.98	0.97	0.97
Reject	0.99	0.98	0.98

Table 4 Implementation of different models for checking effectiveness of our system

Model/classifier	Training accuracy	Validation accuracy
InceptionV3	90.16%	88.60%
DenseNet 201	92.66%	90.19%
CVT	91.67%	89.33%
VGG19	90.44%	88.76%
ResNet50	92.67%	91.88%
ResNet101	91.63%	89.97%
VGG16	92.27%	90.17%
ResNetV2	91.63%	89.77%
ViT	96.37%	92.64%
SENet	96.87%	91.99%
ViT-SENet(Proposed)	**99.87%**	**93.87%**

Bold indicates the proposed framework

the performance of various models on a shared dataset. In general, the confusion matrix is a comprehensive tool used to assess the performance of classification models.

We assessed the outcomes of the ROC curve for our three classes. In machine learning model evaluation, the trade-off between the true positive rate and false positive rate across various threshold settings is evaluated by measuring the ROC curve's value. As a whole, the model's efficacy is evaluated by the AUC-ROC, with a higher value suggesting better ability to distinguish between classes. We have obtained remarkable outcomes in terms of our system's performance metrics. The ripe class achieved an AUC of 0.98, whereas both the unripe class and the class of reject samples performed exceptionally well, with AUC scores of 0.99. The AUC values provide a complete measure of the effectiveness of our approach across various classes. Figure 10 displays the ROC curve results for all three classes, demonstrating the model's ability to distinguish and perform well in these particular categories.

We have achieved substantial progress in the field of tomato fruit classification, namely in differentiating between three critical classifications: ripe, unripe, and reject. The outcomes of our endeavors have produced outstanding results, demonstrating a remarkable level of precision in forecasting the characteristics of tomato fruit classification. The entire visualization of our findings comprises the original images and predicted images. Figure 11 effectively represents the core of our accomplished tomato fruit classification, displaying the anticipated outcomes with clarity and accuracy. This figure illustrates the predicted indicators for the tomato fruit classification and emphasizes the feasibility of accurate prediction. This accomplishment signifies noteworthy progress in the field, providing a useful contribution to the precise classification to ensure food safety and security.

Table 5 Training, Validation, and Test Accuracy of proposed model Using fivefold Cross-Validation

Fold	Train accuracy	Valid accuracy	Test accuracy
1	92.68%	87.84%	85.12%
2	96.77%	94.19%	96.44%
3	97.45%	95.39%	96.44%
4	99.78%	97.38%	98.69%
5	99.97%	98.23%	99.90%

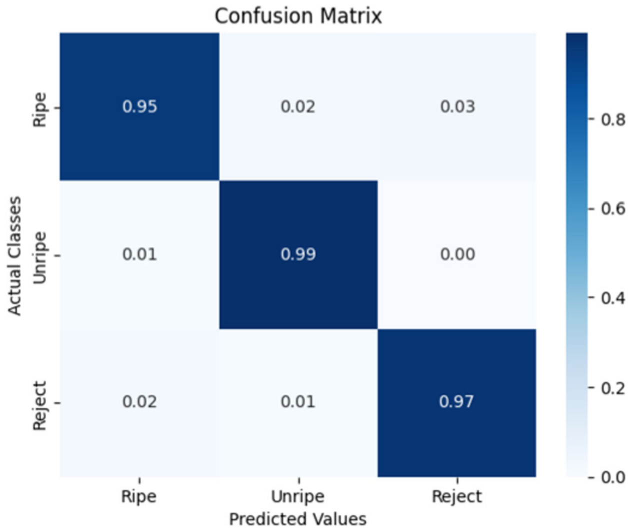


Fig. 9 The confusion matrix shows how the fruit ripeness classification model performs in three different classes

4.3 Comparison of existing and proposed system

After carefully comparing our suggested model to other research, we have discovered that our model is more efficient and accurate than previous work. Specifically, we compared our model against Hu et al. [33], Gutierrez et al. [34], Verma et al. [35], Rangarajan et al. [36], Karthik et al. [37], Da Costa et al. [38], Sun et al. [39], Tran et al. [40], Luna et al. [41], Liu et al. [42], Bendary et al. [43], Yamamoto et al. [44], and Vazquez et al. [45]. The existing works mainly focused on the classification of tomato fruits, with two or three classes. Table 6 compares the details of previous publications and the proposed model. Here, we present the dataset details, including classes, image count, train-test-validation ratios, augmentation status, and the architectures used along with their test accuracies. Our proposed model achieved the highest test accuracy compared to existing research, demonstrating that our system is more accurate, faster, and reliable for tomato fruit

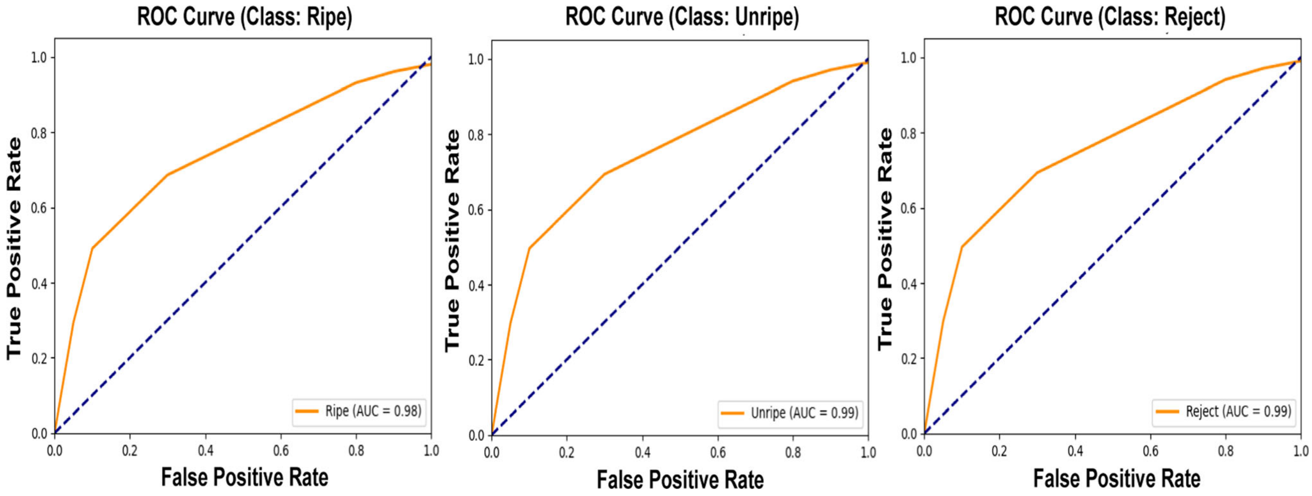
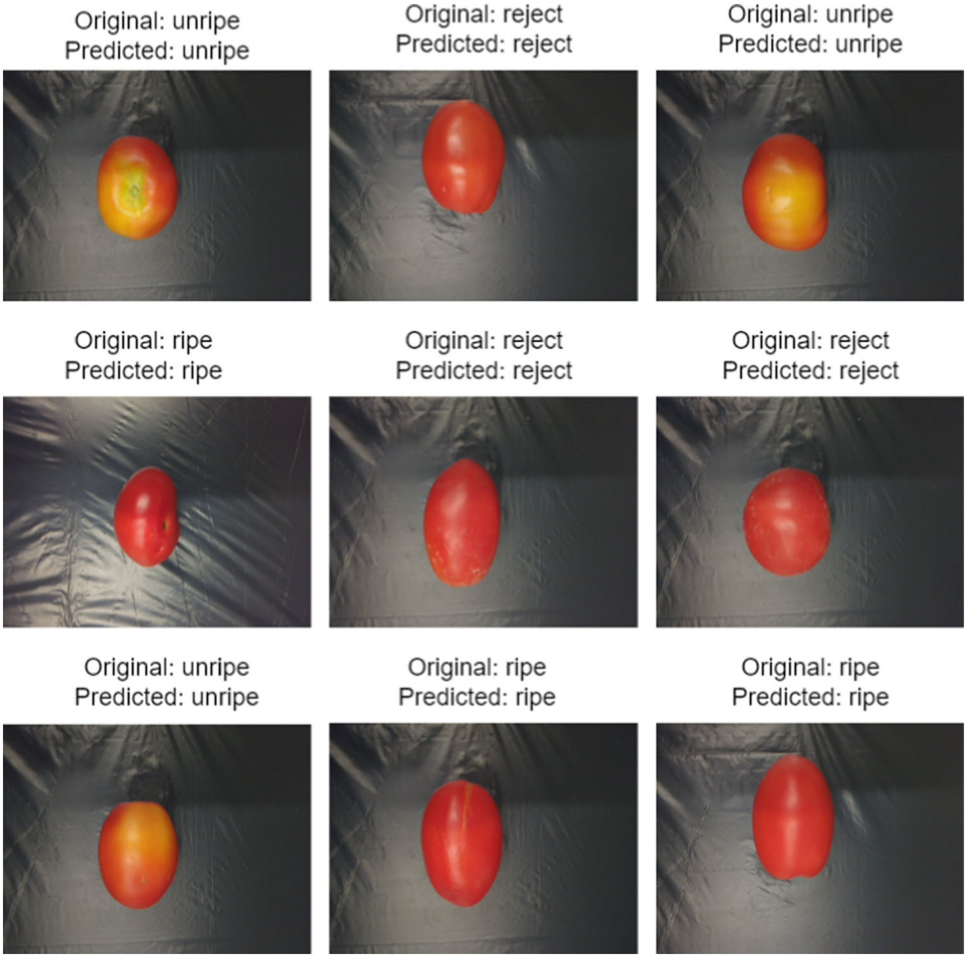


Fig. 10 Displaying our model's receiver operating characteristic (ROC) curve and showing how well it performs for binary classification tasks at various thresholds

Fig. 11 Using our proposed model, displaying the projected images will demonstrate the model's capacity to provide predictions or outputs based on input images



classification. In this comparison, we are unable to directly compare our model's results with published findings, as our model is novel, and no prior research has been conducted using this approach.

5 Discussion

This study introduces a automated tomato classification method that is both fast and efficient. We have devised a hybrid technique called *ViT-SENet* for this implementation. The hybrid *ViT-SENet* model represents an innovative approach that merges the strengths of two distinct architectures: ViT and SENet. The proposed model represents a novel integration of the ViT architecture with the SENet, combining the strengths to enhance performance in vision-based tasks. The vision transformer (ViT) is based on a self-attention mechanism that allows the model to capture long-range dependencies in an image by treating patches of an image as input tokens. This is a significant departure from traditional convolution-based architectures, as it avoids the locality bias in favor of a global image understanding. The ViT uses multi-head self-attention layers to compute pairwise relationships between patches. It then applies position encoding to maintain spatial information, allowing it to capture complex image features. On the other hand, SENet is a lightweight attention mechanism designed to recalibrate channel-wise feature responses by learning a set of attention weights. These weights are generated using a squeeze operation, which aggregates channel-wise statistics, followed by an excitation operation that adjusts the channels accordingly, allowing the model to focus on the most informative features. In the hybrid ViT-SENet model, the SENet is integrated into the ViT framework and applied after the multi-head self-attention layers to recalibrate the learned features from the self-attention module. This combination enables the model to benefit from both the global contextual awareness of ViT and the refined channel-wise attention of SENet, resulting in improved feature representation and performance. The hybrid approach enhances the ViT's ability to focus on critical image regions while maintaining its capability to learn complex long-range dependencies, leading to more accurate predictions and improved generalization across diverse vision tasks.

However, our system has limitations as well. The architecture we have created is quite effective for categorization tasks. However, employing segmentation, natural language processing (NLP), and other tasks may lead to a decline in performance and a drop in system efficiency. Our study introduces a strong and accurate approach for categorizing tomato fruits, which is thoroughly described in our results section. This cutting-edge technique ensures rapid and effective identification of tomato fruits, differentiating between those that are in good condition and those that may pose a risk.

We achieved exceptional accuracies during the evaluation of our model. During training, the model achieved an accuracy of 99.87%, while the validation phase accuracy was 93.87%. These metrics highlight our model's effectiveness and indicate a highly accurate classification system. Additionally, we assessed the testing accuracy using k-fold cross-validation. The highest test accuracy was obtained on fold-5, reaching an impressive 99.90%. This result underscores the reliability and precision of our system during the testing phase. Beyond accuracy, we also

Table 6 Comparison of the findings from the existing model with the proposed model

Ref	Classes	Images	Train/test/valid	Augmentation	Architecture	Test accuracy
[33]	-	800	75/25/0	No	FRCNN	95.5%
[34]	3	N/A	N/A	yes	FRCNN & SSD	N/A
[35]	3	2342	80/20/0	No	AlexNet	89.69%
[36]	3	13,262	N/A	yes	AlexNet	97.49%
[37]	4	1,20,000	N/A	yes	deep CNN	97%
[38]	2	43,843	50/25/25	yes	ResNet50	94.6%
[39]	3	5624	80/20/0	No	FRCNN	90.5%
[40]	3	571	80/20/0	yes	Ensemble	91%
[41]	3	4,923	80/20/0	yes	FRCNN	95.75%
[46]	2	1,594	N/A	No	Customized	91.26%
[42]	2	2385	N/A	No	MobileNetv2	91.32%
[43]	1	-	-	No	SVMs	84.80%
[44]	3	2000	-	No	BBS	88%
[45]	2	1424	80/20/0	No	SVM	88%
Proposed	**3**	**12000**	**70/15/15**	**Yes**	**ViT-SENet**	**99.90%**

Bold indicates the proposed framework

measured other performance metrics, including precision, recall, F1 score, ROC curve, and confusion matrix. These metrics further demonstrate the robustness and effectiveness of our system.

Implementing this system came with several challenges that ultimately contributed to its robustness. Firstly, developing the novel model was particularly demanding. Designing the neural network, along with its functions and layers, required intricate planning and advanced techniques. This was one of the most complex and challenging aspects of the work. Secondly, we addressed the limitations of our dataset, which contained relatively few images, by employing advanced augmentation techniques. These techniques significantly expanded the dataset, enhancing its diversity and enabling the model to generalize better. Thirdly, we undertook hyperparameter tuning and k-fold cross-validation to optimize the model's performance. These rigorous evaluation and optimization processes were both time-intensive and technically challenging but proved crucial for achieving reliable results.

Through the utilization of this technology, we can rapidly distinguish and segregate tomatoes that are suitable for eating from those that should be rejected, so guaranteeing the safety and excellence of the food. Precise categorization of tomato fruits is crucial for multiple reasons. First and foremost, it guarantees quality control by discerning between various tomato types, hence assisting in the marketability and meeting the preferences of consumers. Additionally, it improves food safety by facilitating the identification of particular varieties that are susceptible to specific diseases or pollutants. Furthermore, accurate categorization aids in the optimization of supply chain management by promoting streamlined distribution and minimizing inefficiencies. Furthermore, accurate categorization aids in focused breeding endeavors, maximizing agricultural output and resilience.

6 Conclusion and future plan

In this research, we proposed the *ViT-SENet-Tom* framework that employed a hybrid *ViT-SENet* model. This approach allowed for the rapid and precise categorization of tomato fruits into three groups: ripe, unripe, and reject. Our classification method was precise, with a validation accuracy of 93.87% and a training accuracy of 99.87%. In addition, our study demonstrated a maximum accuracy rate of 99.90%, confirming the dependability and efficiency of our approach. This study emphasized the capacity of machine learning to greatly better the classification of tomato fruits, which has implications for enhancing food security and safety.

In future, we aim to address this limitation in our study to enhance the system's implementation in future work. Additionally, we would include a wider variety of tomato cultivars, with the goal of improving the safety and security of these fruits. We would implement sophisticated procedures and substantially expand our dataset as part of our strategy. Furthermore, we plan to incorporate generative adversarial networks (GANs) into our system to generate realistic synthetic images. This approach will enhance our dataset and serve as a valuable tool for future experiments.

Funding No funding support is available.

Data availability Data are available in a publicly accessible repository that does not issue DOIs. Publicly available datasets were analyzed in this study. These data can be found here: https://www.kaggle.com/datasets/nexuswho/tomatofruits, accessed on 07 03 2024.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Payne A, Walsh K, Subedi P, Jarvis D (2014) Estimating mango crop yield using image analysis using fruit at 'stone hardening'stage and night time imaging. Comput Electron Agri 100:160–167
2. Mondal K, Sharma N, Malhotra S, Dhawan K, Singh R (2004) Antioxidant systems in ripening tomato fruits. Biologia Plantarum 48:49–53
3. Syahrir WM, Suryanti A, Connsynn C (2009) Color grading in tomato maturity estimator using image processing technique. In: 2009 2nd IEEE international conference on computer science and information technology, IEEE, pp 276–280
4. Ma Z, Xue J-H, Leijon A, Tan Z-H, Yang Z, Guo J (2016) Decorrelation of neutral vector variables: theory and applications. IEEE Trans Neural Netw Learn Syst 29(1):129–143
5. Pieczywek PM, Cybulska J, Zdunek A, Kurenda A (2017) Exponentially smoothed fujii index for online imaging of biospeckle spatial activity. Comput Electron Agri 142:70–78

6. Swapno SMR, Nobel SN, Islam MB, Haque R, Meena V, Benedetto F (2024) A novel machine learning approach for fast and efficient detection of mango leaf diseases. In: 2024 IEEE 3rd international conference on computing and machine intelligence (ICMI), IEEE, pp 1–7
7. Nobel SN, Swapno SMR, Islam MR, Safran M, Alfarhood S, Mridha M (2024) A machine learning approach for vocal fold segmentation and disorder classification based on ensemble method. Sci Rep 14(1):14435
8. Nobel SN, Swapno SMR, Ramachandra A, Shajeeb HH, Islam MB, Haque R (2024) Hybrid CNN LSTM approach for sentiment analysis of bengali language comment on facebook. In: 2024 international conference on integrated circuits and communication systems (ICICACS), IEEE, pp 1–8
9. Nobel SN, Swapno SMR, Islam MB, Meena V, Benedetto F (2024) Performance improvements of machine learning-based crime prediction, a case study in bangladesh. In: 2024 IEEE 3rd international conference on computing and machine intelligence (ICMI), IEEE, pp 1–7
10. Bappi MBR, Swapno SMR, Rabbi MF (2024) Skin cancer disease detection using MCD-GRU: a deep learning approach. In: 2024 6th international conference on electrical engineering and information and communication technology (ICEEICT), IEEE, pp 445–450
11. Bappi BR, Swapno SMR, Chhabra G, Kaushik K, Nobel SN, Islam MB (2023) Deep learning based tuberculosis and pneumonia disease detection using CNN. In: 2023 7th international conference on image information processing (ICIIP), IEEE, pp 670–676
12. Rahman S, Haque R, Swapno SMR, Islam MB, Nobel SN, et al (2023) Deep learning-based left ventricular ejection fraction estimation from echocardiographic videos. In: 2023 international conference on evolutionary algorithms and soft computing techniques (EASCT), IEEE, pp 1–6
13. Bappi MBR, Swapno SMR, Akhter S, Rabbi MF (2024) Deploying hybrid VGG19-BiGRU model for kidney disease segmentation. In: intelligent systems conference, Springer, pp 47–61
14. Bappi MBR, Swapno SMR, Rabbi MF (2023) Deploying densenet for cotton leaf disease detection on deep learning. In: International conference on trends in electronics and health informatics, Springer, pp 485–498
15. Swapno SMR, Chhabra G, Kaushik K, Nobel SN, Islam MB, Shahiduzzaman M (2023) An adaptive traffic signal management system incorporating reinforcement learning. In: 2023 annual international conference on emerging research areas: International conference on intelligent systems (AICERA/ICIS), IEEE, pp 1–6
16. Swapno SMR, Nobel SN, Ramachandra A, Islam MB, Haque R, Rahman MM (2024) Traffic light control using reinforcement learning. In: 2024 international conference on integrated circuits and communication systems (ICICACS), IEEE, pp. 1–7
17. Hasan S, Dhakal A, Siddiqua MA, Rahman MM, Islam MM, Chowdhury MAR, Swapno S, Nobel S (2024) Analyzing musical characteristics of national anthems in relation to global indices. arXiv preprint arXiv:2404.03606
18. Imran MA, Swapno SMR, Chhabra G, KaushiK K, Mahi MAA, Islam MB, Haque R (2024) Iot-Enabled smart manhole management system for real-time status, water level, and gas detection. In: 2024 international conference on intelligent systems for cybersecurity (ISCS), IEEE, pp 01–07
19. Oliveira AN, Bolognini SRF, Navarro LC, Delafiori J, Sales GM, Oliveira DN, Catharino RR (2023) Tomato classification using mass spectrometry-machine learning technique: a food safety-enhancing platform. Food Chem 398:133870
20. Lu T, Han B, Chen L, Yu F, Xue C (2021) A generic intelligent tomato classification system for practical applications using densenet-201 with transfer learning. Sci Rep 11(1):15824
21. Alajrami MA, Abu-Naser SS (2020) Type of tomato classification using deep learning. Int J Acad Pedagogical Res (IJAPR) 3:21–25
22. Pereira LFS, Barbon S Jr, Valous NA, Barbin DF (2018) Predicting the ripening of papaya fruit with digital imaging and random forests. Comput Electron Agric 145:76–82
23. Chakraborty S, Shamrat FJM, Billah MM, Al Jubair M, Alauddin M, Ranjan R (2021) Implementation of deep learning methods to identify rotten fruits. In: 2021 5th international conference on trends in electronics and informatics (ICOEI), IEEE, pp 1207–1212
24. Worasawate D, Sakunasinha P, Chiangga S (2022) Automatic classification of the ripeness stage of mango fruit using a machine learning approach. Agri Eng 4(1):32–47
25. Su F, Zhao Y, Wang G, Liu P, Yan Y, Zu L (2022) Tomato maturity classification based on se-yolov3-mobilenetv1 network under nature greenhouse environment. Agronomy 12(7):1638
26. Laykin S, Alchanatis V, Fallik E, Edan Y (2002) Image-processing algorithms for tomato classification. Trans ASAE 45(3):851
27. Phan Q-H, Nguyen V-T, Lien C-H, Hou MT-K, Le N-B et al (2023) Classification of tomato fruit using yolov5 and convolutional neural network models. Plants 12(4):790
28. Tomato fruits dataset. https://www.kaggle.com/datasets/nexuswho/tomatofruits. [Accessed 07-March-2024]
29. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017) Attention is all you need. 31st Conference on neural information processing systems (NIPS 2017), Long Beach, CA. pp 6000–6010
30. Alayrac J-B, Recasens A, Schneider R, Arandjelović R, Ramapuram J, De Fauw J, Smaira L, Dieleman S, Zisserman A (2020) Self-supervised multimodal versatile networks. Adv Neural Inf Process Syst 33:25–37
31. Huang J, Ren L, Zhou X, Yan K (2022) An improved neural network based on senet for sleep stage classification. IEEE J Biomed Health Inf 26(10):4948–4956
32. Aditya A, Zhou L, Vachhani H, Chandrasekaran D, Mago V (2021) Collision detection: An improved deep learning approach using senet and resnext. In: 2021 IEEE international conference on systems, Man, and Cybernetics (SMC), IEEE, pp 2075–2082
33. Hu C, Liu X, Pan Z, Li P (2019) Automatic detection of single ripe tomato on plant combining faster r-cnn and intuitionistic fuzzy set. IEEE Access 7:154683–154696
34. Gutierrez A, Ansuategi A, Susperregi L, Tubío C, Rankić I, Lenža L (2019) A benchmarking of learning strategies for pest detection and identification on tomato plants for autonomous scouting robots using internal databases. J Sensors 2019:1–15
35. Verma S, Chug A, Singh AP (2020) Application of convolutional neural networks for evaluation of disease severity in tomato plant. J Discrete Math Sci Cryptogr 23(1):273–282
36. Rangarajan AK, Purushothaman R, Ramesh A (2018) Tomato crop disease classification using pre-trained deep learning algorithm. Procedia Comput Sci 133:1040–1047
37. Karthik R, Hariharan M, Anand S, Mathikshara P, Johnson A, Menaka R (2020) Attention embedded residual cnn for disease detection in tomato leaves. Appl Soft Comput 86:105933
38. Da Costa AZ, Figueroa HE, Fracarolli JA (2020) Computer vision based detection of external defects on tomatoes using deep learning. Biosyst Eng 190:131–144

39. Sun J, He X, Ge X, Wu X, Shen J, Song Y (2018) Detection of key organs in tomato based on deep migration learning in a complex background. Agriculture 8(12):196
40. Tran T-T, Choi J-W, Le T-TH, Kim J-W (2019) A comparative study of deep cnn in forecasting and classifying the macronutrient deficiencies on development of tomato plant. Appl Sci 9(8):1601
41. De Luna RG, Dadios EP, Bandala AA (2018) Automated image capturing system for deep learning-based tomato plant leaf disease detection and recognition. In: TENCON 2018-2018 IEEE region 10 conference, IEEE, pp 1414–1419
42. Liu J, Pi J, Xia L (2020) A novel and high precision tomato maturity recognition algorithm based on multi-level deep residual network. Multimed Tools Appl 79:9403–9417
43. El-Bendary N, El Hariri E, Hassanien AE, Badr A (2015) Using machine learning techniques for evaluating tomato ripeness. Expert Syst Appl 42(4):1892–1905
44. Yamamoto K, Guo W, Yoshioka Y, Ninomiya S (2014) On plant detection of intact tomato fruits using image analysis and machine learning methods. Sensors 14(7):12191–12206
45. Vazquez DV, Spetale FE, Nankar AN, Grozeva S, Rodríguez GR (2024) Machine learning-based tomato fruit shape classification system. Plants 13(17):2357
46. Liu J, Pi J, Xia L (2020) A novel and high precision tomato maturity recognition algorithm based on multi-level deep residual network. Multimed Tools Appl 79:9403–9417

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

S M Masfequier Rahman Swapno[1] · S. M. Nuruzzaman Nobel[1] · Md Babul Islam[2] · Pronaya Bhattacharya[3] · Ebrahim A. Mattar[4]

✉ Pronaya Bhattacharya
pbhattacharya@kol.amity.edu

✉ Ebrahim A. Mattar
ebmattar@uob.edu.bh

S M Masfequier Rahman Swapno
masfequier.cse.bubt@gmail.com

S. M. Nuruzzaman Nobel
smnuruzzaman712@gmail.com

Md Babul Islam
babulcseian@gmail.com

[1] Department of CSE, Bangladesh University of Business and Technology, Dhaka 1216, Bangladesh

[2] Department of Computer, Modeling, Electronic, and System Engineering, UNICAL, Rende, Italy

[3] Department of Computer Science and Engineering, Amity School of Engineering and Technology, and Research and Innovation Cell, Amity University, Kolkata, West Bengal, India

[4] Robotics - Cybernetic, College of Engineering, University of Bahrain, Manama, Bahrain