SvedbergOpen
DISSEMINATION OF KNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: https://www.svedbergopen.com/

Research Paper Open Access

Lightweight Deep Learning Approaches For Internal Mango Defect Detection Using Image Segmentation

Chandrakant Bhange[1], Dr. Shobha Nikam[2]

[1]Department of E&TC, AISSMS's Institute of Information Technology, SavitribaiPhule Pune Uiversity, Pune, India,
Email: bhange.chandrakant@gmail.com

[2]Department of E&TC, AISSMS'S College of Engineering, SavitribaiPhule Pune University, Pune,India
Email: shobhs.nikam32@gmail.com

Abstract

Precision in identifying internal blemishes in mangoes is one of the most significant issues that cannot be overcome by automated quality evaluation of the products, especially export based grading whereby manual evaluation is inefficient, subjective, and destructive. This research will be conducted with the aim of coming up with and testing lightweight deep learning methods to achieve reliable and internal mango defect detection without compromising on accuracy or complexity of the computations. Internal image dataset of mangoes, comprising of both normal mangoes and defective ones, was created by self and images were preprocessed to remove noise, normalization, and resizing to segment defect-prone areas using the OpenCV. GLCM contrast, correlation, energy, homogeneity, local binary patterns, as well as texture features were extracted and data augmentation methods were used to enhance model generalization. An out-of-the-box custom CNN was used and tested against lightweight transfer learning-based architectures such as MobileNetV2-Lite, EfficientNetB0-Lite, NASNet-Lite, DenseNet121-Lite and ResNet50-Lite when trained under the same conditions. Experimental data indicates that segmentation is a performance boosting strategy in all the models. MobileNetV2-Lite obtained the highest results with a score of 98% accuracy, 97% precision, 99% recall, and a F1-score of 98% with segmentation and only 2.34 million parameters and model size of 8.93 MB. Other designs like NASNet-Lite and DenseNet121-Lite attained 93 and 85 percent accuracy respectively whereas the baseline CNN attained 85 percent. The results confirm that lightweight transfer learning models yielded with segmentation yield an efficient, scalable, and computationally efficient method of internal mango defect detection, thus is adequate and applicable to the real-time agricultural quality inspection systems.

Keywords: Internal mango defect detection; Lightweight deep learning; Image segmentation; Transfer learning, MobileNetV2;Agricultural quality inspection; Computer vision.

1. Introduction

Mango is an economically important tropical fruit, which is highly exported based on the high nutritional value, sensory and commercial value; but internal flaws like spongy tissue, internal breakdown, bruising, and fungal infections profoundly affect the quality but it is not visible to the eye making it a significant challenge to post-harvest quality inspection [1], [2]. Traditional mango grading methods are time-intensive, subjective, labor-intensive, and manual inspection and destructive testing, which are not frequent to use in large-scale export business where uniformity of quality and international standards are the order of the day [3]. Non-destructive methods of inspection such as machine vision and picture-based analysis thus have been the subject of an increasing amount of focus as a potential alternative to automated fruit quality evaluation [4]. However, even with the recent developments in surface defects detection, the internal defects are still not reliably detected since their differences are complicated with texture variations, light imbalance, and faint marks that are hidden in the fruit tissue [5].

The recent advancements in deep learning have greatly changed the way agricultural images are analyzed with automated feature learning and better results in classification than the traditional machine learning techniques which relied on handcrafted features [6]. CNNs, specifically, have proven to be incredibly successful in

classifying fruits, detecting diseases, and identifying quality grading processes through the derivation of hierarchical representations of space directly out of image data [7]. Nevertheless, there exist challenges when implementing deep CNN models to the real agricultural systems: high computational cost, large model size and inapplicability to embedded or edge-based systems frequently present in packing houses and sorting plants [8]. In response to such limitations, the effectiveness of lightweight deep learning models and the transfer learning methods has provided a competitive alternative that is comparatively more accurate with greatly lower parameters and memory needs [9]. Other models include MobileNet, EfficientNet, NASNet, DenseNet, and ResNet, which make use of depthwise separable convolutions, compound scaling, and feature reuse, to trade off performance and efficiency, which makes them especially appealing to agricultural problems with resource constraints [10].

Simultaneously, the preprocessing and segmentation of images are very crucial in improving the visibility of internal defects through isolating regions of interest and reducing the background noise which enhances the feature discrimination and model generalization [11]. Descriptors that are based on texture such features (Gray Level Co-occurrence Matrix (GLCM) and local binary patterns) also supplement the deep features by incorporating minor statistical changes related to internal abnormalities [12]. It is driven by these developments that the current study aims to design and analyze lightweight deep learning methods in internal mango defect detection through image segmentation and transfer learning. An exogenous dataset of internal mango images is used, which undertakes a significant amount of preprocessing, segmentation, and augmentation to make sure that it is robust. Several architectures that are lightweight are systematically compared under equal training conditions to determine the effects of segmentation on the accuracy, precision, recall, and F1-score of classification. This study will help to offer a practical and scalable solution to automated internal mango quality inspection by focusing on computational efficiency and detection performance to facilitate export-grade sorting systems and care about intelligent agriculture technology.

The paper has the following major Contributions:

- ✓ Created a self-generated internal mango image database with preprocessing, segmentation and augmentation that was used to achieve credible defect detection performance.
- ✓ Comparisons and implementation of a variety of lightweight deep learning and transfer learning models with the same training conditions.
- ✓ Obtained substantially higher accuracy with image segmentation, with 98% accuracy with MobileNetV2-Lite model.
- ✓ Developed an effective, low-complexity architecture that would be applicable in real-time as well as export-focused systems of mango quality inspection.

2. Related Work

In the recent years, tremendous developments in non-destructive quality measurement of fruits have been seen, as computer vision and machine learning methods are developed to address the shortcomings of manual inspection systems [13]. The initial researches concerning the assessment of the quality of mangoes were based on external characteristics like size, color and surface texture using classical image processing and handcrafted characteristics, which demonstrated the weak ability to detect internal defects concealed under the peel [14]. In order to overcome this shortcoming, scientists considered statistical texture measures, such as Gray Level Co-occurrence Matrix (GLCM), local binary patterns, and wavelet texture features, which have shown a moderate success in the detection of internal tissue abnormalities in combination with more classical classifiers like support vectors machines and k-nearest neighbors [15], [16]. These methods were however very sensitive to feature engineering and generally did not generalize well to different illumination and fruit maturity conditions [17].

The advent of the deep learning and especially Convolutional Neural Networks (CNNs) provided the turning point in the agricultural image analysis through the ability to automatically learn features directly off of the raw image data which resulted in enhanced resilience and classification capabilities [18]. A number of the studies have effectively implemented CNN-based models in detecting fruit defects, disease, and quality grading,

indicating that it is better than the traditional machine learning pipelines [19]. However, the majority of them utilized deep and computationally intensive architectures, which are less appropriate to deploy in real-time using these constrained resources in the agricultural setting [20]. In a move to reduce this dilemma, lightweight and effective CNN architectures have been of growing interest. MobileNet-based models which apply depthwise separable convolutions have been reported to demonstrate competitive performance in fruit defect classification problems at a significantly lower number of parameters and a much lower inference time than some of their counterparts [21].

In the same way, EfficientNet also presented compound scaling measures to balance network depth, width, and resolution with a high accuracy-efficiency trade-off on various agricultural datasets [22]. Neural architecture search has also been applied to defect detection as neural architecture designs, which provide optimized structural designs with lower computational costs, have been proposed as NASNet architectures [23]. The variants of the two networks, called denseNet and ResNet, also led to performance improvement through amplifying the flow of the gradient and the sharing of features, which allowed the detection of subtle defect patterns in complex agricultural images [24]. Simultaneously, image processing and segmentation methodologies have been universally identified as important elements that enhance accuracy of defect detection. Segmentation Threshold-based segmentation Threshold-based segmentation is demonstrated to be capable of increasing the visibility of the defects through isolating the regions of interest and acquiring the irrelevant background information [25]. Region-based masking Region-based masking has also been shown to be effective in improving the visibility of the defects by isolating and suppressing the irrelevant information about the region of interest [25].

Morphological operations Morphological operations Morphological operations have also been found to increase the visibility of the defects by isolating and suppressing the irrelevant background information [25].

Experiments combining segmentation and the deep learning models record significant improvements in the classification performance, especially in internal defect detection tasks due to the lack of clear boundaries of defects [26]. Although this has happened, in comparison to the other lightweight architectures, across the singular framework of preprocessing and segmentation, there is still a lack of comparative analysis of the various lightweight architectures, particularly when it comes to detecting internal mango defects using the self-generated datasets. In addition, little research has been done to systematically evaluate the trade-off between the detection accuracy, model sizes, and computational efficiency in export-oriented mango quality inspection. Filling these gaps, the current work will expand upon the available literature and combine the concept of segmentation-based preprocessing with lightweight transfer learning models and provide a detailed comparative study and thus offer a scalable and production-ready solution to the problem of automated internal mango defect detection.

Table 1. Summary of Related Work on Internal Fruit and Mango Defect Detection

Fruit / Dataset	Methodology Used	Preprocessing / Segmentation	Model Type	Key Findings
Mango	Classical image processing	Color normalization	SVM	Effective for surface defects only
Mango	Texture feature analysis	GLCM-based extraction	KNN	Limited performance for internal defects
Apple	Handcrafted features	Noise filtering	SVM	Moderate accuracy, poor scalability
Citrus	Wavelet + texture features	Histogram equalization	ANN	Sensitive to illumination variations
Mango	Statistical texture analysis	Manual ROI selection	Random Forest	Required extensive feature tuning
Mixed fruits	Deep CNN	Basic resizing	CNN	Improved accuracy over ML methods
Mango	Deep learning	Data augmentation	CNN	High accuracy but computationally heavy
Apple	Deep CNN	Image normalization	VGGNet	Large model size, slow inference
Mango	Lightweight CNN	Minimal preprocessing	MobileNet	Reduced complexity with good accuracy
Banana	Transfer learning	Segmentation-assisted	EfficientNet	Balanced accuracy and efficiency
Mixed fruits	Neural architecture	Standard preprocessing	NASNet	Optimized architecture, high accuracy

	search			
Mango	Hybrid deep learning	Segmentation + augmentation	DenseNet / ResNet	Improved internal defect detection

3. Methodology

The proposed solution had addressed the creation of an effective and light-weight deep learning model to detect internal mango defects precisely with image segmentation and transfer learning. They had been used to ensure domain relevance and realistic evaluation by using a self-created internal image database of mangoes with both normal and defective samples. At the beginning, the mango images were read with OpenCV and were provided with systematic preprocessing, such as resizing, noise detection, normalization, and segmentation to improve the internal defects and minimize the background disturbance. Segmentation on threshold-based isolation of regions of interest had been performed and the segmented results sent to feature learning and classification. GLCM-based contrast, correlation, energy, homogeneity, and local binary patterns were some of the texture features that had been previously extracted to reveal subtle internal structural variations. During training, data augmentation methods like rotation, translation, zooming and flipping had been applied to enhance robustness and generalization. The baseline custom CNN had already been implemented and tested with several lightweight transfer learning architectures, which are MobileNetV2-Lite, EfficientNetB0-Lite, NASNet-Lite, DenseNet121-Lite, and ResNet50-Lite, all trained under the same settings using binary cross-entropy loss and the Adam optimizer. To balance the performance and learning efficiency of a pre-trained model, partial freezing of models and fine-tuning had been implemented. The confusion matrices and classification metrics had shown that the segmentation-improved lightweight models, especially MobileNetV2-Lite, had reached a higher level of accuracy and a much lower level of computational complexity which proved the efficiency of the proposed solution in terms of being useful in mango quality inspection in practice.

3.1 Data Collection and Pre-processing

The data consisted of a self-constructed list of internal mango pictures in normal and defective status. The images were obtained locally and stored and then loaded into a Python environment that was used to analyze them. The dataset was to facilitate internal defects since it captured various texture variation and structural anomalies associated with the export quality mango testing.

a. Image Reading

Mango images were read and coded in numerical matrices to be processed.

Equation (1): Image representation

$$I(x, y, c) \in \mathbb{R}^{\{H \times W \times C\}}$$

Equation (2): Grayscale conversion

$$Ig(x, y) = 0.299R + 0.587G + 0.114B$$

These equations convert raw image data into numerical form suitable for mathematical operations and learning models.

Fig. 2.Healthy and Defective Mango Image Samples

Fig. 2 illustrates images of healthy and defective internal mangoes which are representative and the images are used in training and testing the proposed defect detection models based on the evident texture, color, and structural differences.

b. Image Resizing

Images were scaled to a constant spatial resolution so that they would have similar input dimensions.

Equation (3): Resizing function

$$Ir = f(I, M, N)$$

Equation (4): Bilinear interpolation

$$Ir(x', y') = \sum_i \sum_j I(i,j) \cdot w(i,j)$$

Resizing conserved spatial consistency and facilitated convolutional processing.

c. Noise Removal

Gaussian filtering was applied on the noise to increase the visibility of defects.

Equation (5): Gaussian kernel

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

Equation (6): Filtering operation

$$If = I * G(\sigma)$$

This blocked noise in high-frequency whilst conservation of structural details was observed.

d. Normalization

The intensity of the pixels was normalized in order to stabilize the training and enhance convergence.

Equation (7): Minimum maximum normalization

$$In = \frac{If - Imin}{Imax - Imin}$$

Equation (8): Normalized pixel range

$$In \in [0, 1]$$

These factors have influenced how people in mid-Atlantis understand the environment.

e. Segmentation

Segmentation to identify defect prone areas by threshold based masking.

Equation (9): Thresholding rule

$$S(x,y) = 1, if\ In(x,y) \geq T$$
$$S(x,y) = 0, otherwise$$

Equation (10): Segmented image

$$Is = In \odot S$$

This highlighted inner abnormal areas and enhanced feature learning and classification accuracy.

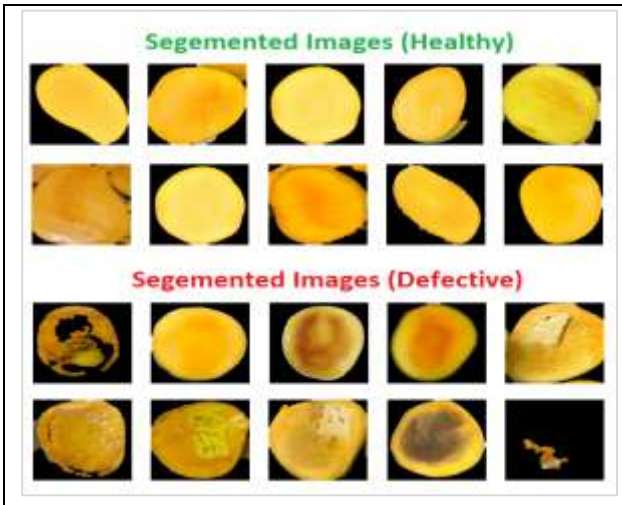

Fig. 3.Segmented Healthy and Defective Mango Images

The graph 3 shows the samples of healthy and defect mango, which are segmented in specific areas of interest, which are clearly isolating the regions of interest and gives internal texture abnormalities which enhances feature extracting and classification accuracy.

3.2 Feature Extraction

The image feature extraction was done to extract the discriminative texture features with internal mango defects. GLCM was calculated on grayscale images with a set pixel distance and orientation, where the statistical analysis was performed to identify the relationship of spatial gray levels. Contrast measured local intensity variations, correlation, the intensity of pixel dependency, energy that reflected the uniformity of texture and homogeneity measured the closeness of pixel distributions all of which were determined in the GLCM. Also, local binary pattern descriptors were obtained to encode local texture structures and micro-patterns. These attributes had a good ability to describe the presence of subtle internal tissue irregularities, which cannot just be detected using the raw pixel values. The features obtained added value to the deep learning representations, as the deep learning representations offer weak statistical texture features, and thus the overall defect discrimination capability of the proposed framework is improved.

Let the pre-processed grayscale image be denoted as I(x,y), where x and y represent pixel coordinates.

1. Gray Level Co-occurrence Matrix (GLCM)

The GLCM P(i,j) represents the probability of occurrence of pixel intensities i and j at a fixed spatial relationship:

$$P(i,j) = \left(\frac{1}{N}\right)\Sigma x, y$$

where N is the normalization factor ensuring Σ P(i,j) = 1.

2. Contrast

Contrast measures local intensity variations and highlights defect-induced texture changes:

$$Contrast = \sum(i-j)2P(i,j)$$

3. Correlation

Correlation quantifies the linear dependency between neighboring pixel intensities:

$$Correlation = \Sigma i,j \left[\frac{(i - \mu i)(j - \mu j)}{\sigma i\, \sigma j}\right] P(i,j)$$

Where μ and σ represent the mean and standard deviation of pixel intensities.

4. Energy

Energy represents texture uniformity and repetition patterns:

$$Energy = \sum_{i,j} P(i,j)^2$$

5. Homogeneity

Homogeneity measures the closeness of pixel distributions to the diagonal of the GLCM:

$$Homogeneity = \Sigma i,j \left[\frac{P(i,j)}{1 + |i - j|}\right]$$

6. Local Binary Pattern (LBP)

LBP encodes local texture by thresholding neighboring pixels:

$$LBP(x,y) = \sum_{n=0}^{N-1} s(In - Ic)\, 2^{n}$$

whereIc is the center pixel, In are neighboring pixels, and

$$s(x) = 1, if\ x \geq 0$$

$$0, if\ x < 0$$

7. Feature Vector Formation

All extracted features are combined into a single descriptor:

$$F = [Contrast, Correlation, Energy, Homogeneity, LBP]$$

This feature space gives a discriminative and succinct representation of internal mango texture patterns to classify defects.

3.3 Data Augmentation

The method of data augmentation was used to enhance the model generalization and reduce overfitting due to small dataset size. The virtual images were dynamically obtained in real time during the training in a way that controlled the geometric transformations. The variability in orientation was solved by random rotations of the range of -20 to +20 degrees, and positional changes were made by horizontal and vertical adjustment up to 10 per cent. Zoom variability with scale within a 15 range of variations explained scale variations and horizontal flipping expanded viewpoint variability. A split of 30% was used to make sure that there was no performance bias. These augmentation plans added variety to the data but did not change the defect semantics so the models were able to learn robust and invariant features that ended up giving a better classification and stability when the mango samples were changed.

Table 2. Augmented images are generated programmatically during training

Parameter	Value	Description
rotation_range	20	Randomly rotates images within a range of ±20 degrees
width_shift_range	0.1	Horizontally shifts images by up to 10% of the width
height_shift_range	0.1	Vertically shifts images by up to 10% of the height
zoom_range	0.15	Randomly zooms images in/out by up to 15%
horizontal_flip	True	Randomly flips images horizontally
validation_split	0.3	Reserves 30% of data for validation

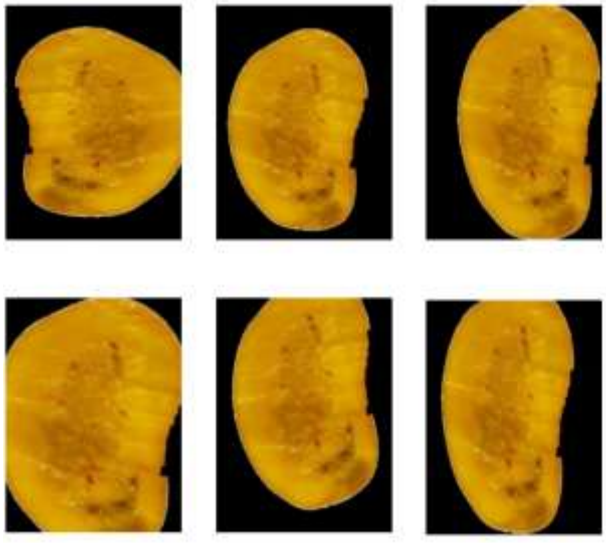

Fig. 4. Augmented Mango Image Samples

The image 4 demonstrates various augmented images of a mango image demonstrating their changes in orientation and appearance which increase the data heterogeneity and enhance model resilience.

4. Model Approach

4.1 Convolutional Neural Network (CNN)

This was a custom convolutional neural network used as a baseline model. It had a stacked convolutional architecture composed of feature extraction, pooling layers composed of spatial reduction and fully connected dense layers composed of classification. This model learnt features on its own and served as a point of reference in performance.

ALGORITHM 1: Custom CNN for Internal Mango Defect Detection

Input: Preprocessed mango image I

Output: Class label Y ∈ {Normal, Defective}

Step 1: Input Layer

$X0 \leftarrow I$

Step 2: Convolution Operation

$Xl = ReLU(Wl * Xl-1 + bl),\ l = 1,2,\ldots,L$

whereWl and bl are convolution weights and bias.

Step 3: Pooling

$Pl = MaxPool(Xl)$

Step 4: Fully Connected Layer

$F = ReLU(Wf \cdot Flatten(PL) + bf)$

Step 5: Classification

$\hat{Y} = Softmax(Wc \cdot F + bc)$

Step 6: Loss Computation

$L = -\Sigma\, y \log(\hat{Y})$

Step 7: Update Parameters using Adam Optimizer

4.2 MobileNetV2-Lite (Proposed)

The transfer learning of MobileNetV2-Lite was done on an ImageNet pre-trained backbone to obtain high accuracy at a low cost in terms of a computation. The separable convolutions and the inverted residual blocks were depthwise allowing the extraction of features efficiently. Mango images were sorted into normal and defective, with custom classification layers being added and adjusted to provide the optimal classification of these mangoes.

ALGORITHM 2: MobileNetV2-Lite (Proposed)

Input: Segmented mango image I

Output: Class label Y

Step 1: Load ImageNet pre-trained MobileNetV2 backbone

Step 2: Depthwise Separable Convolution

Depthwise: $Zd = Wd * X$

Pointwise: $Zp = Wp \cdot Zd$

Step 3: Inverted Residual Block

$Yl = Xl + F(Xl)$ if dimensions match

Step 4: Global Average Pooling

$G = GAP(YL)$

Step 5: Fully Connected Classification Layer

$\hat{Y} = \text{Softmax}(W \cdot G + b)$
Step 6: Binary Cross-Entropy Loss
$L = -[y \log(\hat{Y}) + (1 - y) \log(1 - \hat{Y})]$
Step 7: Fine-tune selected layers using Adam

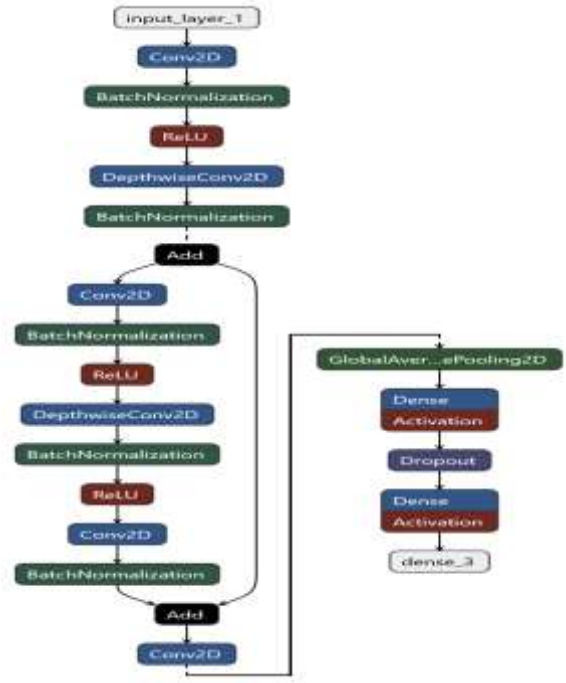

Fig. 5. MobileNetV2 Lite Model Summary Architecture

4.3 EfficientNetB0-Lite

EfficientNetB0-Lite used the scaling of compounds to stabilize network depth, width and input resolution. ImageNet pre-trained weights were applied and the last few layers were trained to detect internal mango defects. This model was able to strike an effective balance between the performance in the classification and the amount of resources required by working under restricted resources.

ALGORITHM 3: EfficientNetB0-Lite
Input: Preprocessed mango image I
Output: Class label Y

Step 1: Load pre-trained EfficientNetB0
Step 2: Compound Scaling
Depth: $d = \alpha^{\varphi}$, Width: $w = \beta^{\varphi}$, Resolution: $r = \gamma^{\varphi}$
Step 3: MBConv Block
$X_{l+1} = X_l + \text{MBConv}(X_l)$
Step 4: Global Average Pooling
$G = \text{GAP}(X_L)$
Step 5: Classification
$\hat{Y} = \text{Softmax}(W \cdot G + b)$
Step 6: Compute Binary Cross-Entropy Loss
Step 7: Update trainable parameters

4.4 NASNet-Lite

The design of the architecture was made sure by NASNet-Lite, which was based on the architectures acquired by neural architecture search. Binary mango defect classification using customized fully-trained top layers was done with the help of pre-trained weights. The model used the same preprocessing and training, allowing to use it fairly with reduced computational complexity.

ALGORITHM 4: NASNet-Lite

Input: Preprocessed mango image I
Output: Class label Y
Step 1: Load NASNet pre-trained architecture
Step 2: Normal Cell Computation
$N_l = f(N_{l-1}, N_{l-2})$
Step 3: Reduction Cell for spatial downsampling
Step 4: Feature Aggregation
F = Concatenate(All cell outputs)
Step 5: Global Average Pooling
G = GAP(F)
Step 6: Classification Layer
$\hat{Y} = \text{Softmax}(W \cdot G + b)$
Step 7: Optimize using Adam

4.5 DenseNet121-Lite

DenseNet121-Lite used a dense connection between the layers to facilitate the re-use of features and better gradient flow. Subtle variations in internal texture that correlate with defects in mangoes were detected using this architecture. Higher layers fine tuning permitted successful adjusting with the inner mango image set with a better classification of stability.

ALGORITHM 5: DenseNet121-Lite
Input: Segmented mango image I
Output: Class label Y
Step 1: Load DenseNet121 with pre-trained weights
Step 2: Dense Block Computation
$X_l = H_l([X_0, X_1, \ldots, X_{l-1}])$
where [·] denotes feature concatenation
Step 3: Transition Layer
$T_l = \text{Pool}(\text{Conv}(X_l))$
Step 4: Global Average Pooling
G = GAP(XL)
Step 5: Classification
$\hat{Y} = \text{Softmax}(W \cdot G + b)$
Step 6: Compute Binary Cross-Entropy Loss
Step 7: Fine-tune higher layers

4.6 ResNet50-Lite

ResNet50-Lite made use of the residual connections to allow further training of the networks without the problem of vanishing gradient. ImageNet pre-trained weights were implemented and selective fine-tuning did to mango defect locations. The model represented complex features and at the same time it was stable to training and actually had stable detection.

ALGORITHM 6: ResNet50-Lite

Input: Preprocessed mango image I
Output: Class label Y

Step 1: Load ImageNet pre-trained ResNet50
Step 2: Residual Block
Yl = F(Xl, Wl) + Xl
Step 3: Stack Residual Blocks for deep feature learning
Step 4: Global Average Pooling
G = GAP(YL)
Step 5: Fully Connected Layer
$\hat{Y}$ = Softmax(W · G + b)
Step 6: Binary Cross-Entropy Loss Computation
Step 7: Parameter Update using Adam Optimizer

5. Result Analysis

5.1 Confusion Matrix

The confusion matrix of the baseline CNN shows a moderate result of classification of internal mango defects. The model accurately identified 38 faulty samples and 14 healthy samples, which means that it has the reasonable ability to learn raw images. Nevertheless, 5 defective mangoes were wrongly labeled as healthy and 4 healthy samples were wrongly forecasted as defective. These misclassifications point to the CNN weak capacity of recording fine-grained internal texture differences without the extensive feature reuse or the lightweight optimization, as revealed in figure 6 (a). This was indicated by the fact that the false positives and false negatives are relatively higher, indicating sensitivity to noise and inherent looks of classes. This model can be used as a baseline, showing the necessity of segmentation and transfer learning to enhance the discrimination accuracy and robustness in the real quality inspection systems. Figure 6 (b) the EfficientNetB0-Lite confusion matrix would exhibit biased classification patterns, in which all the samples would be classified as defective.

Despite the identification of 43 faulty mangoes in the correct position, the misidentification of all 18 healthy mangoes occurred, which shows that there are zero true negatives. This result implies that the EfficientNetB0-Lite, based on the specified setup, could not gain knowledge about balanced decision boundaries of both classes. The lack of appropriate healthy predictions indicates the overfitting on the salient defects or the lack of adaptation during the fine-tuning. EfficientNetB0 is characterized as an efficient and accurate framework but this finding demonstrates that appropriate segmentation, focus on features, and the choice of training method is essential. It further points to the fact that lightweight architectures need to be carefully tuned to prevent the effect of class imbalance when dealing with defect detection tasks.

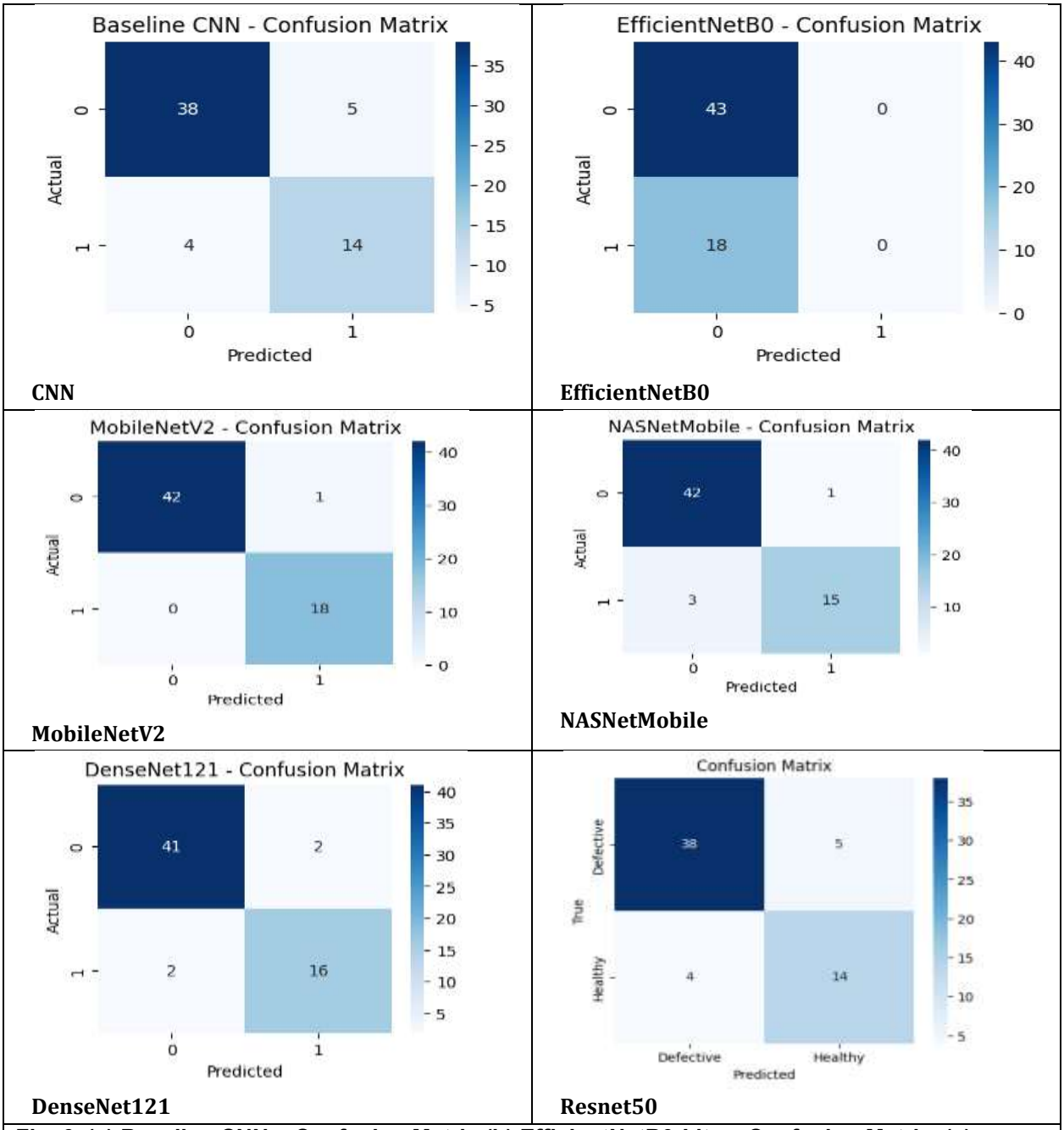

Fig. 6. (a) Baseline CNN – Confusion Matrix (b) EfficientNetB0-Lite – Confusion Matrix (c) MobileNetV2-Lite – Confusion Matrix (d) NASNet-Lite – Confusion Matrix (e) DenseNet121-Lite – Confusion Matrix (f) ResNet50-Lite – Confusion Matrix

Figure 6 (c) the confusion matrix of MobileNetV2-Lite shows a high level of classification performance as compared to other models. The correct classification of mango samples revealed that 42 defective samples and 18 healthy samples were classified correctly and no defective sample was classified as healthy or 18 healthy samples were predicted as defective. This almost perfect division is a sign of great generalization and strength of detecting internal flaws. None of the false positives indicate great confidence in detecting healthy samples which is very important in export-grade quality inspection. The low misclassification proves the power of the segmentation with the help of depthwise separable convolutions and transfer learning. On the whole, MobileNetV2-Lite demonstrated the most stable and balanced performance, which proves its suitability as the best-performing lightweight model in the current research. The NASNet-Lite confusion matrix in figure 6 (d) demonstrates that it has a high classification ability with slight errors. The model has rightly classified the number of defective and healthy samples as 42 and 15 respectively and 1 defective sample was classified as

healthy and 3 healthy samples as defective. The errors show that there were a few mistakes in borderline cases where the internal texture differences are minor. However, most of the predictions proved to be correct, which proves the usefulness of the design with neural architecture search in the process of learning optimal feature representations. The evenly distributed errors indicate constant learning behavior. NASNet-Lite offers an acceptable compromise between complexity and performance and thus it can be a suitable choice in automated internal mango defect systems.

Figure 6 (e) the DenseNet121-Lite confusion matrix indicates the consistent classification where the internal defect patterns are well handled. This model was able to classify 41 defective and 16 healthy samples accurately and only 2 misclassifications in both categories. Thick connections among the layers facilitated reuse of features as well as gradient flow facilitating the capture of fine internal texture variations. The false positive and false negative rates are also relatively low, which means that there is high discrimination. Being slightly less precise than MobileNetV2-Lite, DenseNet121-Lite was also able to remain equally high across both classes. These findings show that dense feature propagation can improve internal defect detection particularly when it is used with segmentation and transfer learning. Figure 6 (f) the ResNet50-Lite confusion matrix indicates a good performance similar to the base CNN that was able to correctly classify 38 defective and 14 healthy samples. Nevertheless, 5 faulty mangoes were not classified correctly as healthy and 4 healthy samples were classified as faulty. The given training configuration showed that the model did not improve significantly as compared to the baseline even though there were residual connections that allowed it to learn more. This indicates that depth by itself is not a guarantee of excellent performance of internal defect detection without an enhancement of lightweight optimization. These findings show that even though ResNet50-Lite can be trained to gain complex representations, it might be slow compared to lighter models to achieve real-time mango quality inspection.

5.2 Accuracy and Loss Curve Analysis

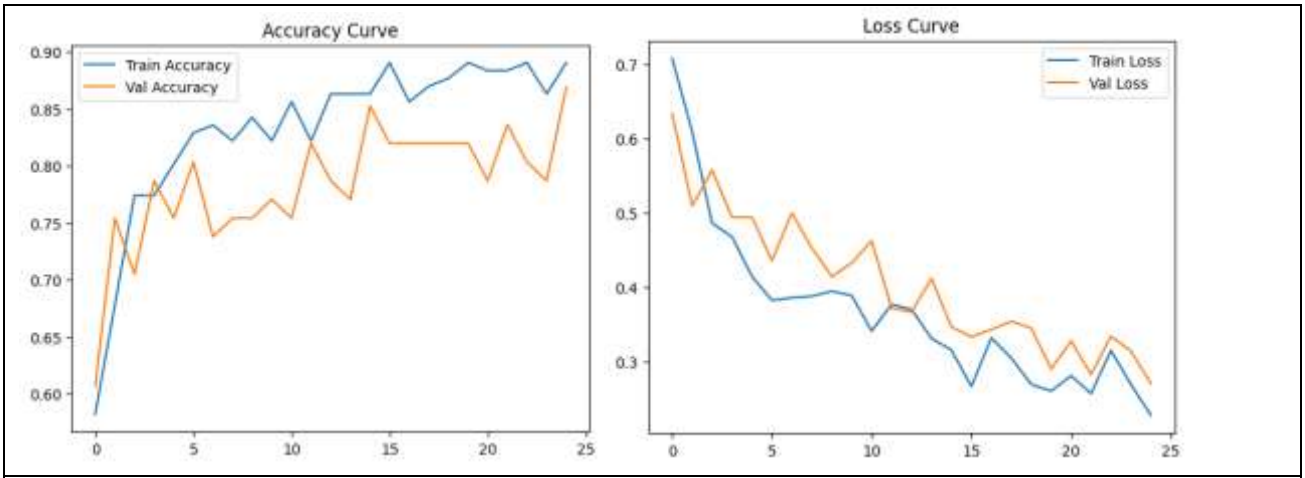

Fig. 7. CNN Model Accuracy and Loss Curve

CNN accuracy curve figure 7 indicates slow learning with average variations, but validation accuracy is a bit lesser. The loss curves decrease consistently, which implies the convergence, however, significant gaps imply poor generalization and the sensitivity to internal defect variability. EfficientNetB0 has a fluctuating accuracy behavior with a high level of validation change. Figure 8 illustrates that the separation between the loss curves is small and therefore there is underfitting and problems with learning discriminative internal features, probably because of the lack of fine-tuning and effective segmentation adaptation. MobileNetV2 shows a high rate of accuracy improvement and a close correspondence between training and validation curves as illustrated in figure 9. Loss reduction becomes more stable, which proves the stability of convergence, successful feature learning, and high generalization of internal mango defect detection. NASNet demonstrates a gradual increase in the accuracy with slight fluctuations between the training and validation curves. The trend

of loss reduction is a sign of consistent optimization as is depicted in figure 10, with minimal differences signifying a realistic generalization with some confusion occurring in the borderline defect cases. DenseNet121 is able to continuously boost accuracy, and validation is well-correlated. The loss curves presented in figure 11 also exhibit gradual loss, which is a sign of efficient gradient flow and successful reuse of features, which allows to detect the presence of minor internal mango defects with reasonable confidence. ResNet50 has inconsistent accuracy curves indicated by figure 12 and moderately converging. The decrease in loss is slow but nonlinear indicating more complexity in the architecture without corresponding performance improvements meaning that value lightweight optimization in the defect detection tasks is essential.

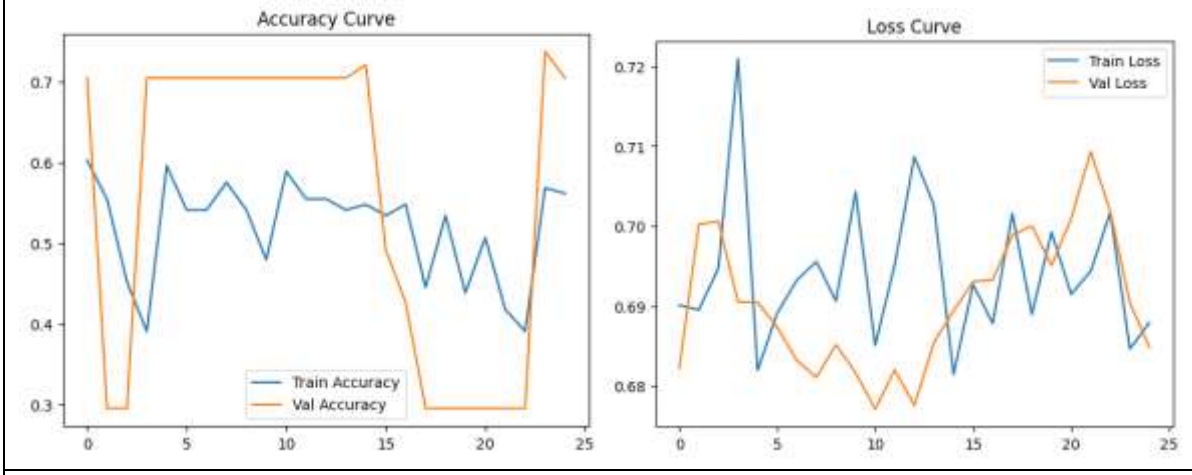

Fig. 8. EfficientNetB0 Model Accuracy and Loss Curve

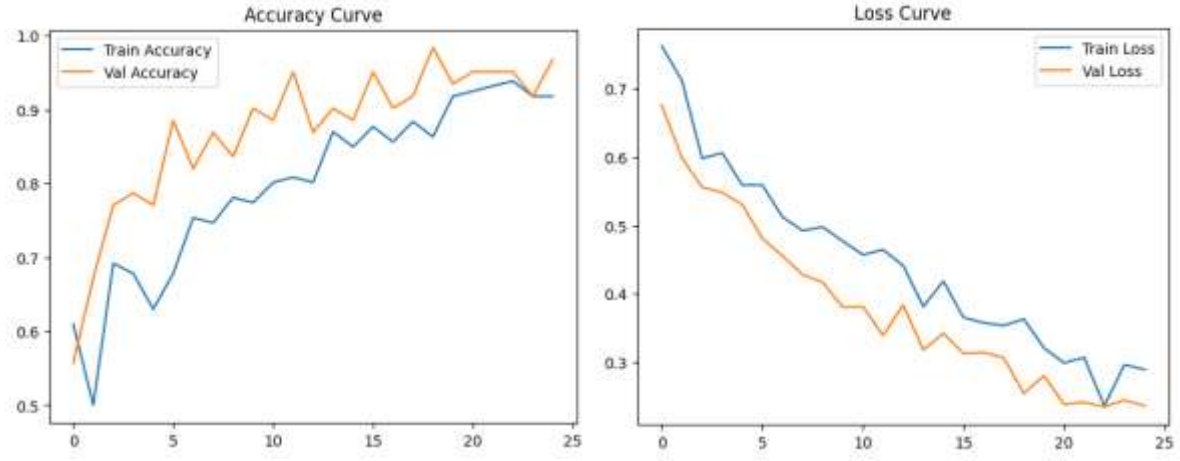

Fig. 9. MobileNetv2 Model Accuracy and Loss Curve

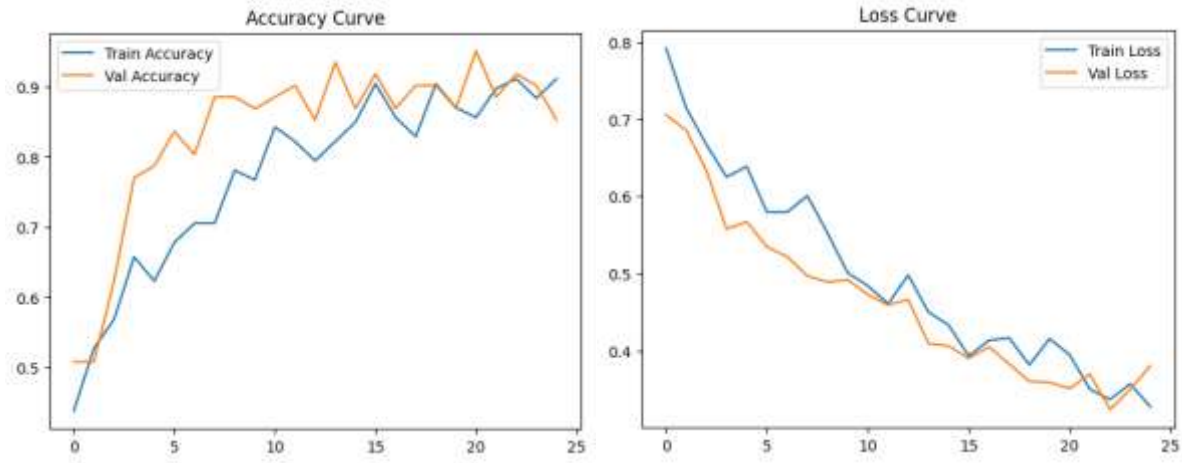

Fig. 10.NASNet Model Accuracy and Loss Curve

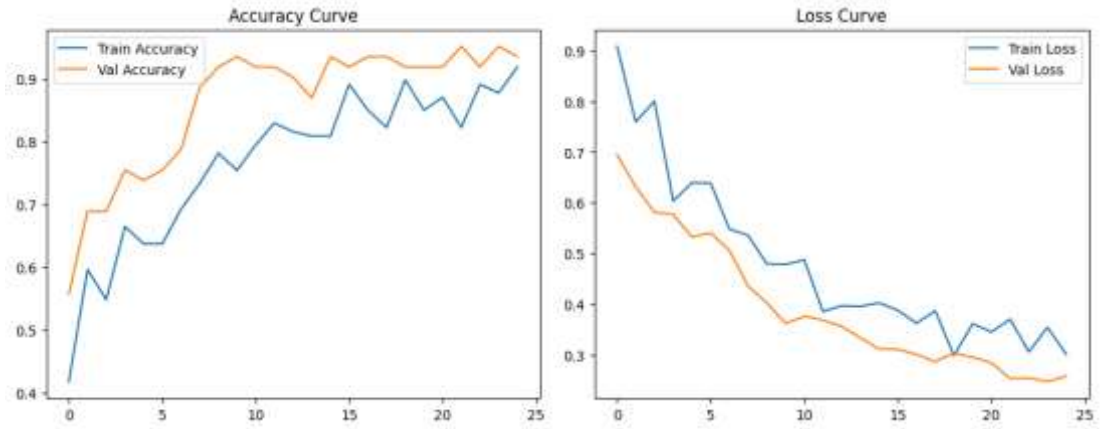

Fig. 11. DenseNet121 Model Accuracy and Loss Curve

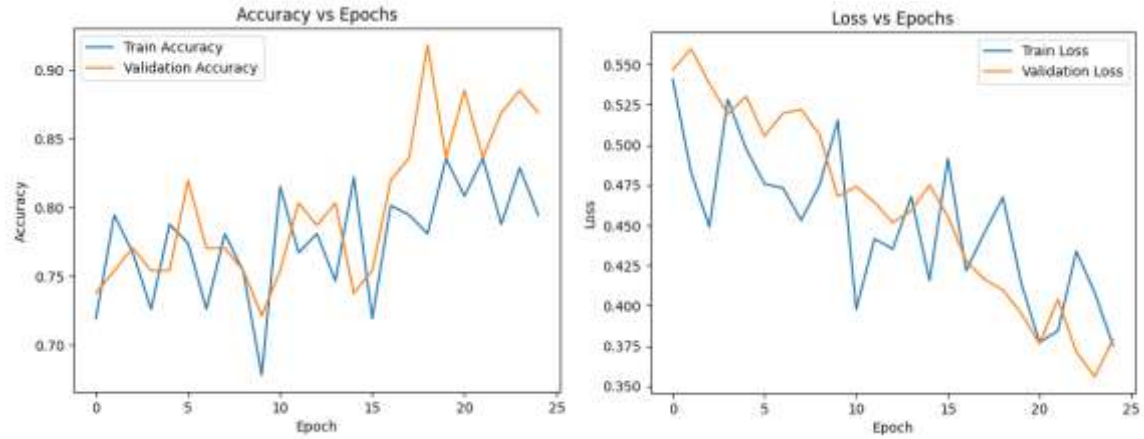

Fig. 12. ResNet50 Model Accuracy and Loss Curve

5.3 Comparisons of different model

The above model comparison results clearly reaffirm that there is a major effect of image segmentation on the performance of internal mango defect detection using all the evaluated architectures. The baseline CNN had an accuracy of 83% without segmentation and this figure rose marginally to 85% with segmentation, which shows that it does not offer much benefit since its features are poorly represented. Without segmentation, EfficientNetB0-Lite did not perform well with an accuracy of 70% and a low precision of 35%, but when segmented, the precision was significantly better (62) and the recall also was better (72), which validates defect localization was improved. The biggest improvement was observed in MobileNetV2-Lite, as the accuracy, precision, recall, and F1-score improved by 80, 78, 83, and 97, respectively, indicating that it has a high generalization and is robust. This also proved to be very beneficial to NASNet-Lite, where it could segment with an accuracy of 93 percent as compared to 73 percent without and significantly higher precision and recall. DenseNet121-Lite proved to be stable, scoring up to 93 percent accuracy and balanced precision recall. ResNet50-Lite went up to 93% accuracy, but the improvements were not as high as for lighter architectures. In general, the findings prove that preprocessing based on segmentation and lightweight transfer learning models, especially MobileNetV2-Lite, can achieve better performance in internal mango defect detection. The table 3 indicates massive performance boosts in all the models as image segmentation is used, particularly in lightweight transfer learning models.

Table 3.Performance Comparison of Deep Learning Models With and Without Image Segmentation for Internal Mango Defect Detection

Model	Accuracy Without Segmentation	Accuracy With Segmentation	Precision Without Segmentation	Precision With Segmentation	Recall Without Segmentation	Recall With Segmentation	F1-Score Without Segmentation	F1-Score With Segmentation
CNN	83	85	83	82	75	83	78	83
EfficientNetB0-Lite	70	70	35	62	50	72	41	68
MobileNetV2-Lite	80	98	78	97	83	99	78	98
NASNet-Lite	73	93	69	94	71	91	70	92
DenseNet121-Lite	80	93	76	92	76	92	76	92
ResNet50-Lite	85	93	82	89	83	89	83	89

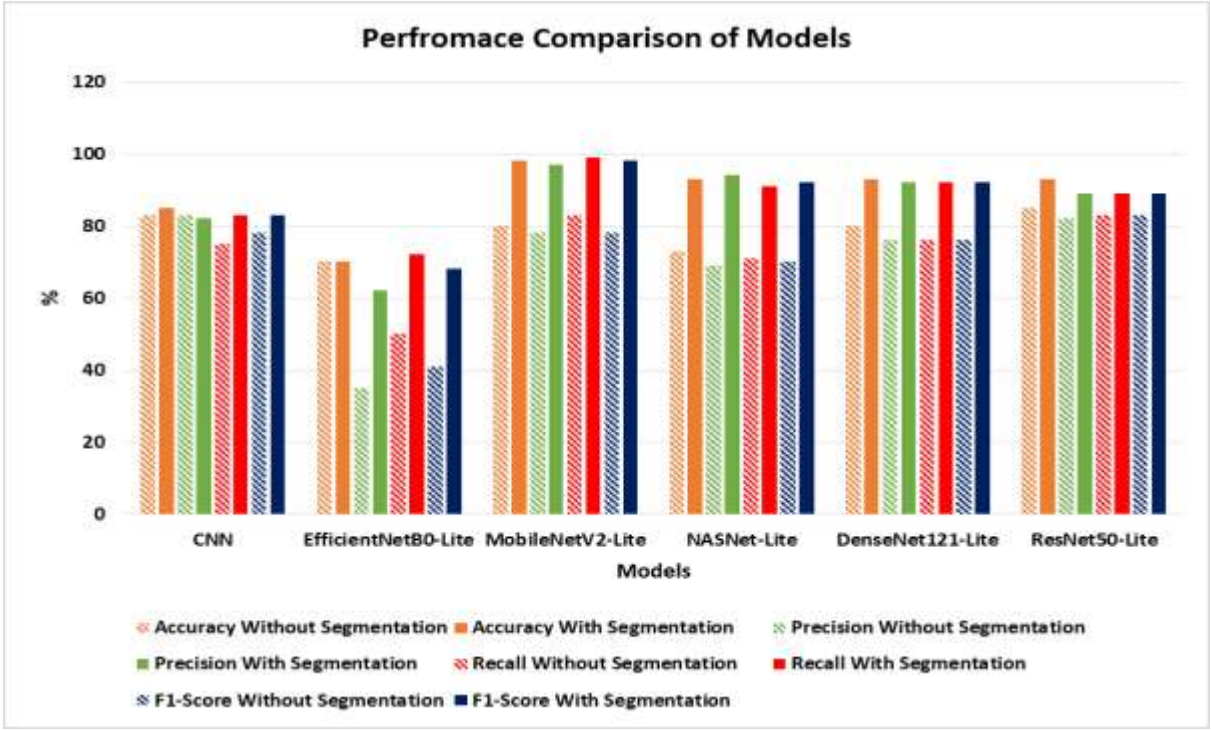

Fig. 13. Representation the Performance comparison of model

Table 4. Model Complexity and Parameter Distribution Analysis of Deep Learning Architectures

Model	Total Params	Trainable Params	Non-Trainable Params	Model Size
CNN	11,169,089	11,169,089	0	42.61 MB
MobileNetV2-Lite	2,340,033	82,049	2,257,984	8.93 MB
EfficientNetB0-Lite	4,131,620	82,049	4,049,571	15.76 MB
NASNet-Lite	4,295,720	82,049	4,049,571	16.39 MB
DenseNet121-Lite	7,103,169	65,665	7,037,504	27.10 MB
ResNet50-Lite	23,850,113	14,712,577	9,137,536	90.98 MB

The table 4 shows a comparative analysis of the model complexity in the number of total parameters, trainable and non-trainable parameters, and the size of the whole model of all the architectures considered. The default CNN has a trainable parameter count of more than 11 million, which makes the model size of 42.61 MB rather large and thus, resource-intensive to compute and hard to use in resource-limited settings. MobileNetV2-Lite, conversely, has the smallest design, and just 2.34 million total parameters, and its model size is 8.93 MB. Most of its parameters cannot be trained because of transfer learning which drastically cuts training overhead with no notable accuracy loss in detection. EfficientNetB0-Lite and NASNet-Lite are moderately complex and have about 4 million parameters and model sizes of about 16 MB, and trade-off efficiency and representational capacity. DenseNet121-Lite has many more parameters because of the dense connectivity and consumes more memory but better features reuse. The most complicated model is ResNet50-Lite, which has 23.85 million parameters and is huge in size of 90.98 MB, and it is not very applicable in lightweight and real-time tasks. On the whole, the comparison shows that lightweight architectures and especially MobileNetV2-Lite would offer better performance and usability when it comes to the internal mango defect detection systems.

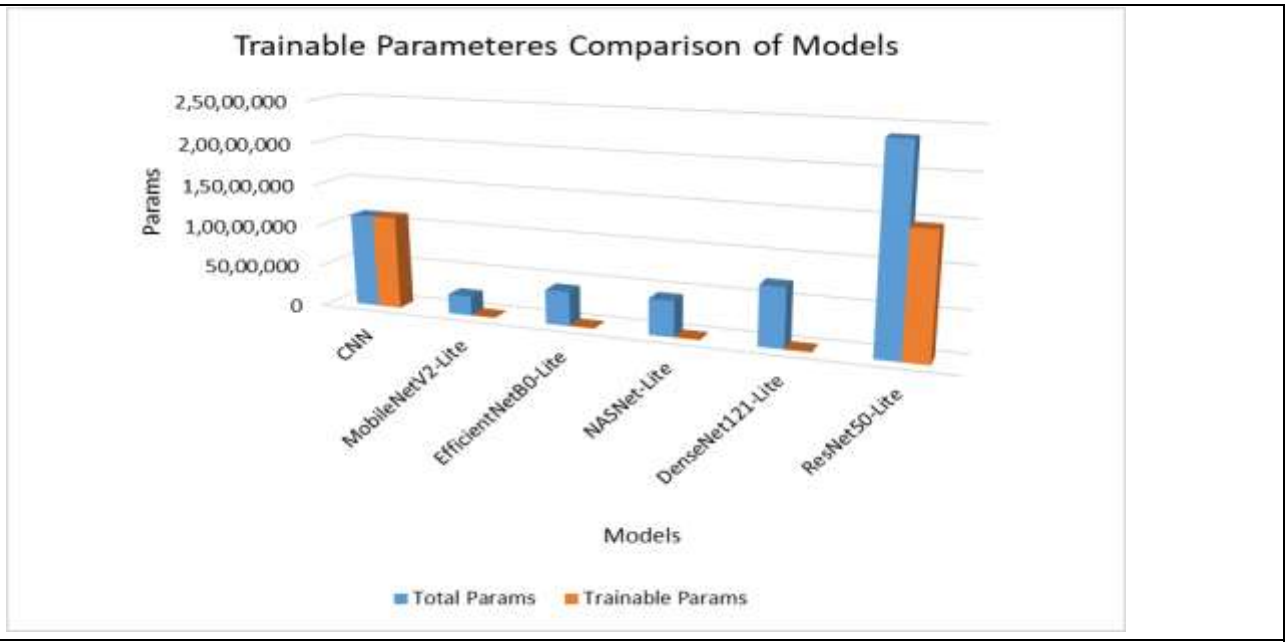

Fig. 14.Comparison of Total and Trainable Parameters across Deep Learning Models

As the figure 14 shows, there is an apparent difference between total and trainable parameters of each model. Lightweight networks (in particular MobileNetV2-Lite) are much smaller with trainable parameters and thus can be easily trained and deployed relative to heavier CNN and ResNet50-Lite networks.

5.4 Predictions

The results of the prediction show the usefulness of the suggested internal mango defect detection model in practice in terms of differentiating healthy and defective samples. In the initial prediction, the model predicted the mango to be HEALTHY with a confidence score of 71.22, meaning that there is the dominant presence of normal internal texture patterns and that there are no abnormalities of significance. The medium-high confidence indicates feature extraction and classification reliable and indicate that the internal structure is similar to learned healthy representations. Conversely, the second prediction was that the mango was DEFECTIVE with a confidence of 50.79, but this number is low, nevertheless, it points to the existence of internal tissue anomalies like discoloration and structural decay. The low score of confidence indicates that the defect features were faint or were partly similar with healthy patterns which indicates the natural challenge of classifying internal defects. These performance results indicate that the model can offer probability levels of confidence in addition to the labeling of a class, which can allow making decisions informed during quality inspection. On the whole, these findings can support the idea that the suggested strategy can be effective to

provide automated quality assessment of mangoes and the use of the confidence values can provide an extra understanding on the reliability of the prediction under the conditions of real-life implementation.

Fig. 15. Prediction Results Showing Healthy and Defective Mango Classification with Confidence Scores

The graph 15 shows results of the predictions of healthy and defective mango samples, with class names and confidence scores, which proves the potential usefulness of the suggested defect detection system and its reliability in making decisions.

6. Conclusion

The paper has provided an efficient and inclusive model of internal mango defect detection with the combination of image segmentation methods and lightweight deep learning and transfer learning models. An internal mango image dataset was created and thoroughly treated with resizing, noise removal, normalization, segmentation, texture feature extraction and data augmentation to increase the visibility of defects and generalization of the models. Decades of experimentation have shown that image segmentation is a critical component in enhancing the overall performance of classification in all models as it isolates defect prone areas as well as minimizes the interference of the background. MobileNetV2-Lite has shown the best performance in the evaluation, with 98% accuracy, 97% precision, 99% recall, and 98% F1-score yet has a relatively small model size of 8.93 MB. By comparison, heavier models like ResNet50-Lite delivered similar accuracy improvements at much higher computational and memory costs, which means that they can only be used in practice when deployed in real-time. The outcomes of prediction also confirmed the strength of the proposed method, with the healthy mango samples being recognized with the confidence of 71.22 and defective samples being recognized with the confidence of 50.79, which clarifies the capability of the model to produce probabilistic information, as well as, identify the class labels. Such confidence scores are especially useful in making the decisions based on the export-oriented quality inspection, where borderline cases have to be thoroughly considered. All in all, the results establish that the lightweight transfer learning models with segmentation provide the best tradeoff between scalability, efficiency and accuracy. The suggested framework is applicable to real-time and non-destructive mango quality assessment systems and may be applied to other types of fruits and agricultural assessment systems, which enhances the development of intelligent and sustainable agri-food processing systems.

References

1. Lin, S., & Chiu, H. (2026). From Quality Grading to Defect Recognition: A Dual-Pipeline Deep Learning Approach for Automated Mango Assessment. Electronics, 15(3), 549. https://doi.org/10.3390/electronics15030549
2. Nithya, R., Santhi, B., Manikandan, R., Rahimi, M., &Gandomi, A. H. (2022). Computer Vision System for Mango Fruit Defect Detection Using Deep Convolutional Neural Network. Foods, 11(21), 3483. https://doi.org/10.3390/foods11213483
3. Wu, Y.-C., Hsu, H.-W., & Chen, S.-W. (2025). Mobile-Based Deep Learning Approach for Multi-Object Aiwen Mango Grade Classification. Electronics, 14(16), 3188. https://doi.org/10.3390/electronics14163188
4. Chaudhary, R. K., Neupane, A., Wang, Z., & Walsh, K. (2025). Mango Quality Assessment Using Near-Infrared Spectroscopy and Hyperspectral Imaging: A Systematic Review. Agronomy, 15(10), 2271. https://doi.org/10.3390/agronomy15102271
5. Xiao, F., Wang, H., Xu, Y., & Zhang, R. (2023). Fruit Detection and Recognition Based on Deep Learning for Automatic Harvesting: An Overview and Review. Agronomy, 13(6), 1625. https://doi.org/10.3390/agronomy13061625
6. Ahmad, I., Khaliq, A., Siddique, B., Gouda, M., Huang, T., Tao, J., &Qiu, Z. (2026). Automated Mango Variety Classification Using Deep Feature Extraction and Machine Learning Classifier Integration. Foods, 15(3), 414. https://doi.org/10.3390/foods15030414
7. Diop, A.; Méot, J.M.; Léchaudel, M.; Chiroleu, F.; Ndiaye, N.D.; Mertz, C.; Cissé, M.; Chillet, M. Impact of Preharvest and Postharvest on Color Changes during Convective Drying of Mangoes. Foods 2021, 10, 490.
8. May, O.S.; Wan Ismail, W.N.A.B.; Jun, C. The King of Fruits Dilemma in Malaysia: Discovering Durian Export Challenges to China. J. Eng. Res. Educ. 2023, 15, 61–75.
9. Sultan, M.T.; Nayyar, A.; Maaz, M.; Noman, A.M. Mango Quality Under Changing Climate Scenario. In Climate Change and Mango Production: Potential Adaptation and Mitigation Options; Naqvi, S.A.H., Ahmad, S., Ahmed, M., Eds.; Springer Nature: Cham, Switzerland, 2025; pp. 109–119.
10. Razak, T.R.; Faizi, N.A.K.; Ismail, M.H.; Fauzi, S.S.M.; Gining, R.A.J. Towards Capturing Mango Grading from Human Experts—A Comprehensive User Study. In Proceedings of the 2021 6th IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE), Kedah, Malaysia, 1–3 December 2021; Volume 6, pp. 1–6.
11. Mangan, T. Draft Report on Mango Farm Survey in Sindh, Pakistan; Monash University: Melbourne, Australia, 2018.
12. Rizwan Iqbal, H.M.; Hakim, A. Classification and Grading of Harvested Mangoes Using Convolutional Neural Network. Int. J. Fruit Sci. 2022, 22, 95–109.
13. Han, B.; Zhang, J.; Almodfer, R.; Wang, Y.; Sun, W.; Bai, T.; Dong, L.; Hou, W. Research on Innovative Apple Grading Technology Driven by Intelligent Vision and Machine Learning. Foods 2025, 14, 258.
14. Peng, W.; Ren, Z.; Wu, J.; Xiong, C.; Liu, L.; Sun, B.; Liang, G.; Zhou, M. Qualitative and Quantitative Assessments of Apple Quality Using Vis Spectroscopy Combined with Improved Particle-Swarm-Optimized Neural Networks. Foods 2023, 12, 1991.
15. Liu, R.M.; Su, W.H. APHS-YOLO: A Lightweight Model for Real-Time Detection and Classification of StrophariaRugoso-Annulata. Foods 2024, 13, 1710.
16. Shao, D.; Cheng, W.; Jiang, C.; Pan, T.; Li, N.; Li, M.; Li, R.; Lan, W.; Du, X. Machine-Learning-Assisted Aroma Profile Prediction in Five Different Quality Grades of NongxiangxingBaijiu Fermented During Summer Using Sensory Evaluation Combined with GC×GC–TOF-MS. Foods 2025, 14, 1714.
17. Gu, Y.; Wu, J.; Guo, Y.; Hu, S.; Li, K.; Shang, Y.; Bao, L.; Hassan, M.; Zhao, C. Grade Classification of Camellia Seed Oil Based on Hyperspectral Imaging Technology. Foods 2024, 13, 3331.
18. Tripathi, P.; Sharma, S. Improving Mango Classification Accuracy with Enhanced Image Preprocessing and Neural Networks. Afr. J. Biomed. Res. 2024, 27, 539–561.
19. Mulukalla, S.S.R.; Kadari, P.R.; Rathod, K.S. External Features Based Grading of Mangoes Using Deep Learning Int. J. Adv. Res. Comput. Commun. Eng. 2025, 14, 229–234.
20. Wijethunga, W.M.U.D.; Shin, M.H.; Jayasooriya, L.S.H.; Kim, G.H.; Park, K.M.; Cheon, M.G.; Choi, S.W.; Kim, H.L.; Kim, J.G. Evaluation of Fruit Quality Characteristics in Irwin Mango Grown via Forcing Cultivation in a Plastic Facility. Hortic. Sci. Technol. 2023, 41, 617–633.
21. Tripathi, M.K. Shivendra Improved Deep Belief Network for Estimating Mango Quality Indices and Grading: A Computer Vision-Based Neutrosophic Approach. Netw. Comput. Neural Syst. 2024, 35, 249–277.
22. Sikder, S.; Islam, M.S.; Islam, M.; Reza, S. Improving Mango Ripeness Grading Accuracy: A Comprehensive Analysis of Deep Learning, Traditional Machine Learnin
23. Raghavendra, A.; Guru, D.S.; Rao, M.K. Mango internal defect detection based on optimal wavelength selection method using NIR spectroscopy. Artif. Intell. Agric. 2021, 5, 43–51.

24. Tayde-Nandure, A., & Shaikh, A. (2026). Advances in Geospatial Technology for Early Detection of Mango Leaf Diseases. International Journal on Advanced Computer Theory and Engineering, 15(1S), 286–293. https://doi.org/10.65521/ijacte.v15i1S.1329
25. Rankhambe , V. S., &Mulajkar , R. M. (2025). A Review of Smart Integrated System for Mango Orchard Management Using IoT. International Journal of Recent Advances in Engineering and Technology, 14(1s), 38–47. https://doi.org/10.65521/intjournalrecadvengtech.v14i1s.243
26. Tai, N. D., Lin, W. C., Trieu, N. M., &Thinh, N. T. (2024). Development of a Mango-Grading and -Sorting System Based on External Features, Using Machine Learning Algorithms. Agronomy, 14(4), 831. https://doi.org/10.3390/agronomy14040831