

Fruit Quality Detection Using Deep Learning Techniques

Thiviya P M.E., Students

Department of Computer Science and Engineering, MRK Institute of Technology , Kattumannarkoil , India

Senthamaraiselvi M, Assistant Professor

Department of Computer Science and Engineering, MRK Institute of Technology , Kattumannarkoil , India

Akilan A, HOD

Department of Computer Science and Engineering, MRK Institute of Technology , Kattumannarkoil , India

Abstract-- Fruit quality assessment is an important task in agricultural production, postharvest processing, export inspection, and retail supply chains. Manual inspection of fruit freshness and spoilage is slow, subjective, and inconsistent, especially when large volumes of fruits must be graded within a short time. This article converts the thesis work into an IEEE-style research article and presents an automated deep learning framework for classifying fruit images into fresh and rotten categories. The proposed approach combines convolutional neural networks (CNNs), spatial pyramid pooling (SPP), finite impulse response (FIR) based preprocessing, stacked sparse denoising autoencoder (SSDAE) feature optimization, and a support vector machine (SVM) classifier. CNN layers extract color, texture, shape, and surface defect features; SPP preserves spatial details without forcing all images into a fixed input size; SSDAE reduces noise and redundancy; and SVM improves the final decision boundary. Experiments reported in the thesis show that the hybrid method achieves 97.25% accuracy, 98.10% precision, 98.98% recall, 98.81% F1-score, 95.62% specificity, and a low prediction error of 0.15.

Index Terms-- Fruit quality detection, deep learning, CNN, spatial pyramid pooling, SSDAE, SVM, image processing, postharvest loss, smart agriculture.

I. INTRODUCTION

A. Background and Motivation

Agriculture is a foundation of economic development and food security. The value of fruit crops depends not only on yield but also on visible quality, freshness, ripeness, nutritional safety, and freedom from surface defects. Fruits such as apples, bananas, oranges, and mangoes are highly perishable, and their quality changes quickly during harvesting, storage, transportation, and retail display. When quality deterioration is not identified early, the supply chain suffers postharvest losses, reduced market price, customer dissatisfaction, and lower export acceptability.

Conventional fruit quality inspection is commonly performed by human graders who visually examine color, size, shape, bruises, fungal infection, spots, and other surface irregularities. Although skilled inspectors can produce useful decisions, manual inspection becomes unreliable when thousands of fruits must be evaluated rapidly. Different inspectors may grade the same fruit differently because of fatigue, lighting conditions, experience, and subjective judgement. Manual grading is also difficult to integrate with smart agriculture and automated sorting lines.

Computer vision and artificial intelligence provide a practical route to reduce these limitations. A fruit image can be captured through a camera or mobile device, processed through image enhancement operations, passed to a trained neural network, and classified within seconds. Deep learning is particularly attractive because it learns hierarchical features directly from image data instead of requiring handcrafted descriptors. For this reason, CNN-based methods have become a dominant approach for fruit quality detection, disease identification, and grading.

The thesis on which this article is based proposes a hybrid deep learning method for fruit quality detection. It first evaluates baseline models such as a manual CNN, LeNet-5, and AlexNet, and then introduces an improved architecture based on CNN-SPP and SSDAE-SVM. The conversion into an IEEE-style article condenses the thesis into a publication-oriented structure while retaining the essential methodology, figures, tables, and experimental findings.

B. Problem Definition

The core problem addressed in this work is binary fruit quality classification from images. Given a fruit image captured from a camera or selected from storage,

the system must determine whether the fruit belongs to the fresh category or the rotten category. This decision should remain reliable despite variations in fruit type, illumination, background, orientation, scale, and surface texture.

Existing CNN-based systems often require a fixed input image size. Resizing images can distort shape information or remove fine defect details. Traditional machine learning methods also depend on handcrafted features and are sensitive to threshold selection. Therefore, a robust system requires automatic feature extraction, the ability to process variable-sized images, noise-resistant feature optimization, and a strong binary classifier.

TABLE I PROBLEM MAPPING AND PROPOSED TECHNICAL SOLUTION

Issue	Effect on Quality Detection	Proposed Solution
Manual inspection	Subjective grading and low throughput	Automated CNN-based image classification
Fixed image size	Loss of spatial details during resizing	Spatial Pyramid Pooling layer
Noise and illumination variation	Unstable feature extraction	FIR filtering and normalization
Overfitting	Poor performance on unseen fruit images	Dropout and denoising autoencoder
Weak decision boundary	Misclassification between fresh and rotten classes	SVM classifier after deep feature extraction

C. Contribution of the Study

The main contribution of the work is the design of a hybrid fruit quality detection architecture that combines the feature learning power of CNNs with the flexible feature aggregation of SPP and the classification strength of SVM. The SSDAE module strengthens the feature representation by learning noise-free encodings, while dropout reduces overfitting during training.

The article presents a clear system architecture, module-level explanation, algorithmic flow, experimental tables, comparative performance analysis, and implementation details.

II. RELATED WORK

Fruit quality detection has evolved from manual visual inspection to machine vision, machine learning, transfer learning, and hybrid deep learning methods. Early approaches relied on color thresholding, segmentation, and handcrafted texture features. Later approaches adopted SVM, GLCM descriptors, HOG features, and clustering. More recent studies use CNNs, ResNet, VGG, YOLO, attention mechanisms, and autoencoder-based feature refinement. The literature review in the thesis covers studies demonstrating a steady increase in accuracy as algorithms become more adaptive and data-driven.

A. Traditional Machine Vision and Machine Learning

Traditional systems typically segment fruit regions from the background and then compute handcrafted descriptors such as color histograms, texture measures, shape moments, and edge features. These features are then classified using SVM, decision trees, or other machine learning methods. Such systems are simple and fast, but they often fail under changing lighting, camera angle, background variation, and complex surface patterns.

Studies based on K-means clustering and SVM showed that color and texture information can distinguish rotten and fresh fruits with reasonable accuracy. Mango defect detection using GLCM features also demonstrated the importance of texture analysis. However, threshold-based segmentation and manually selected descriptors cannot capture the full variability of fruit disease symptoms and spoilage patterns.

B. Deep Learning and Transfer Learning

Deep learning models learn features automatically from raw pixels. CNNs extract low-level edges and color transitions in early layers and increasingly complex surface patterns in deeper layers. Transfer learning further improves performance by adapting a pretrained network such as AlexNet, VGG, ResNet, or DenseNet to a fruit dataset. This reduces training time and improves generalization when the available dataset is limited.

Transfer learning studies using ResNet50 and VGG16 show strong performance for ripeness detection and disease classification. YOLO-based models support real-time detection and industrial sorting, while attention-based ResNet models improve interpretability by focusing on diseased surface regions. Nevertheless, many deep networks still require fixed-size inputs and can lose details during resizing.

C. Hybrid and Robust Feature Learning Approaches

Hybrid models attempt to combine the strengths of multiple algorithms. CNN-SVM models use CNNs as feature extractors and SVM as a discriminative classifier. SSDAE-based models learn robust representations under noise and illumination changes. CNN-SPP models handle variable-sized inputs and preserve important spatial structure. The thesis combines these ideas into an integrated CNN-SPP and SSDAE-SVM solution for fruit quality classification.

The review shows that performance improvement is achieved not only by deeper networks but also by better preprocessing, robust feature aggregation, regularization, and classifier design. Therefore, the proposed model is motivated by the need for a practical and accurate framework rather than simply a deeper CNN.

TABLE II SUMMARY OF RELATED TECHNIQUES FOR FRUIT QUALITY DETECTION

Technique	Main Advantage	Main Limitation
K-means + SVM	Simple and interpretable classification	Sensitive to lighting and thresholds
GLCM + classifier	Useful texture descriptors	Limited representation of complex defects
CNN	Automatic hierarchical feature extraction	Usually needs fixed-size input
Transfer learning	Good accuracy with smaller datasets	May require GPU resources
YOLO-based detection	Real-time object detection and grading	Needs large annotated datasets
CNN-SPP	Handles variable image size	Requires careful integration with classifier
SSDAE-SVM	Noise-resistant features and strong boundary	Training complexity is higher

III. PROPOSED METHODOLOGY

A. System Overview

The proposed system follows a modular workflow beginning with image acquisition and ending with quality prediction. Fruit images are uploaded or captured

through a camera. Preprocessing improves image quality through filtering, normalization, resizing when required, and background handling. CNN layers then extract visual features, and the SPP layer converts the variable-sized feature maps into fixed-length feature vectors. SSDAE refines these vectors by removing noise and redundancy. Finally, SVM classifies the fruit as fresh or rotten and produces a confidence score.

The system architecture indicates the relationship among user interface, image acquisition, feature extraction, classification, and result display modules. This layered architecture is suitable for academic projects because each module can be tested independently and then integrated into a complete end-to-end prototype.

B. Image Acquisition and Dataset Preparation

The input dataset contains fruit images representing both fresh and rotten samples. The thesis focuses on selected fruit types including apple, banana, orange, and mango. The image collection contains variations in color, texture, orientation, and surface condition. These variations are important because a classification model should learn general quality cues rather than memorizing a single fruit pose or background.

Image acquisition may be performed through a webcam, mobile camera, industrial camera, or stored image repository. For laboratory implementation, JPEG and PNG images can be organized into class folders such as Fresh and Rotten. A typical split divides the dataset into training, validation, and testing sets. The training set is used to learn model parameters, the validation set is used for tuning and early evaluation, and the testing set is used for final performance reporting.

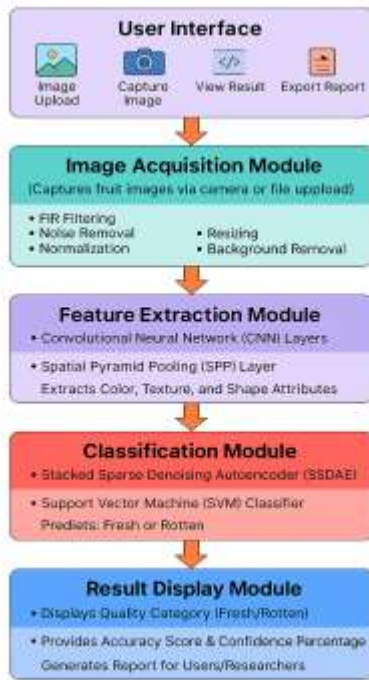


Fig. 1. System architecture overview for fruit quality detection using preprocessing, feature extraction, classification, and result display modules.

TABLE III DATASET CHARACTERISTICS USED IN THE ARTICLE

Parameter	Description
Classification type	Binary classification: Fresh and Rotten
Fruit examples	Apple, banana, orange, and mango
Input format	RGB images collected from camera or file upload
Important visual cues	Color, texture, bruises, fungal marks, spots, shape and surface defects
Training objective	Learn discriminative visual features for quality category prediction
Deployment context	Agriculture laboratories, sorting units, mobile applications and retail inspection



Fig. 2. Apple dataset samples containing fresh and rotten examples used for model learning and testing.



Fig. 3. Banana dataset samples showing quality variation and surface deterioration patterns.



Fig. 4. Orange dataset samples showing visible changes associated with freshness, ripeness, and spoilage.

C. Preprocessing with FIR Filtering

Preprocessing reduces the influence of external conditions and improves the reliability of feature extraction. Fruit images can contain noise, non-uniform illumination, reflections, shadows, and background pixels that do not contribute to quality classification. The thesis proposes the use of a finite impulse response filter to preserve edge quality while reducing noise.

In addition to filtering, normalization scales pixel values into a consistent range, and color space transformation can support more effective segmentation. Resizing is needed for baseline CNN models, but the proposed SPP-based method reduces the dependence on fixed input size. Background removal or masking can further improve attention on the fruit surface region.

TABLE IV PREPROCESSING OPERATIONS AND THEIR ROLES

Operation	Purpose	Expected Benefit
FIR filtering	Remove noise while preserving edges	Cleaner surface features
Normalization	Scale pixel intensity values	Stable neural network learning
RGB to HSV conversion	Separate color and brightness information	Improved segmentation
Background	Reduce	Better focus on

d removal	irrelevant pixels	fruit surface
Data augmentation	Create rotated, flipped and varied samples	Reduced overfitting

D. CNN-SPP Feature Extraction

A CNN contains convolutional layers, activation functions, pooling operations, and fully connected layers. Convolutional filters scan the input image and extract local patterns such as edges, spots, color transitions, and texture irregularities. Deeper layers combine these local features into higher-level representations of bruises, fungal growth, and rotten regions.

The limitation of a conventional CNN is its dependence on fixed input dimensions. When fruit images are resized to match the network input, small defects may become blurred or distorted. Spatial Pyramid Pooling solves this limitation by pooling features at multiple spatial scales and producing a fixed-length representation from feature maps of different sizes. This allows the network to preserve useful spatial information while remaining compatible with a classifier.

$F = SPP(CNN(I))$, where I is the input image and F is the fixed-length feature vector. (1)

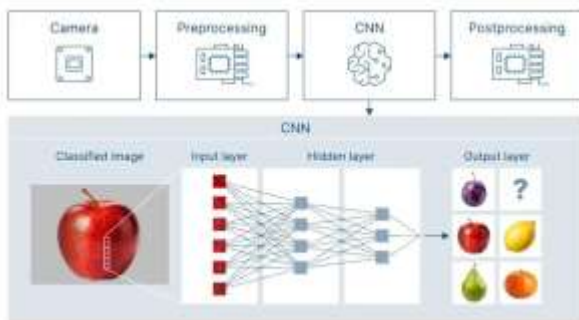


Fig. 5. General CNN-based image processing workflow from camera input to preprocessing, neural feature extraction, and postprocessing.

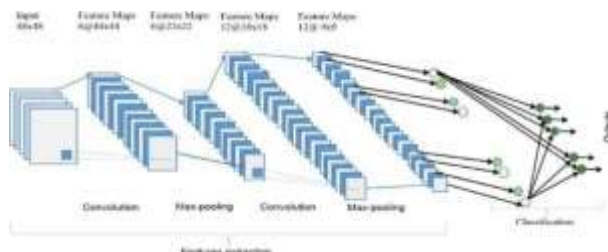


Fig. 6. Convolutional neural network feature extraction layers used to learn hierarchical visual patterns from fruit images.

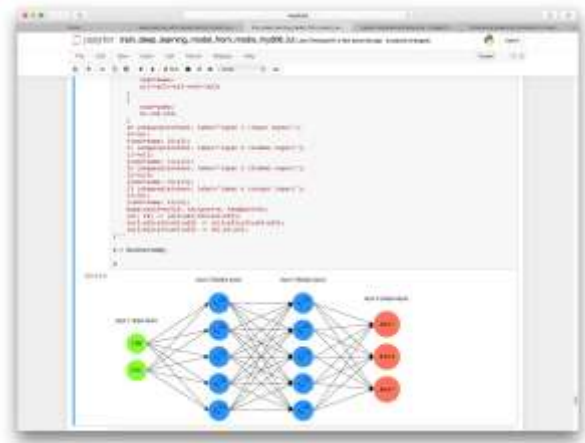


Fig. 7. Illustration of neural network layers used for classification experiments and model visualization.

E. SSDAE Feature Optimization and SVM Classification

The Stacked Sparse Denoising Autoencoder learns compressed and noise-resistant representations of the CNN-SPP feature vector. During denoising, noise is added to the input representation, and the network learns to reconstruct the original feature vector. This encourages the learned representation to capture robust structure rather than noise. Sparsity constraints further encourage compact, meaningful features.

The optimized features are classified using a Support Vector Machine. The SVM constructs a decision boundary that maximizes the separation between fresh and rotten classes. When a radial basis function kernel is used, the SVM can learn nonlinear class boundaries suitable for complex visual patterns. This combination supports high accuracy and stable classification performance.

$y = SVM(SSDAE(F))$, where $y \in \{Fresh, Rotten\}$. (2)

TABLE V HYBRID MODEL COMPONENTS AND FUNCTIONAL ROLES

Component	Role in the System	Reason for Selection
CNN	Extracts visual features from fruit images	Strong feature learning for images
SPP	Converts variable-sized feature maps to fixed vectors	Preserves spatial information
FIR filter	Enhances images and reduces noise	Keeps edge details useful for defect detection
SSDAE	Learns robust and compressed	Improves noise resistance and

	feature representations	generalization
Dropout	Randomly disables neurons during training	Reduces overfitting
SVM	Performs final binary classification	Provides strong decision boundary

IV. SYSTEM DESIGN AND IMPLEMENTATION

A. Overall System Design

The system design contains two complementary paths: the training path and the prediction path. During training, labeled fruit images are processed, features are extracted, and the model parameters are learned. During prediction, a new fruit image is captured or uploaded, processed through the same feature pipeline, and classified by the trained model. The design is suitable for a graphical user interface where the user uploads an image and receives the predicted quality category.

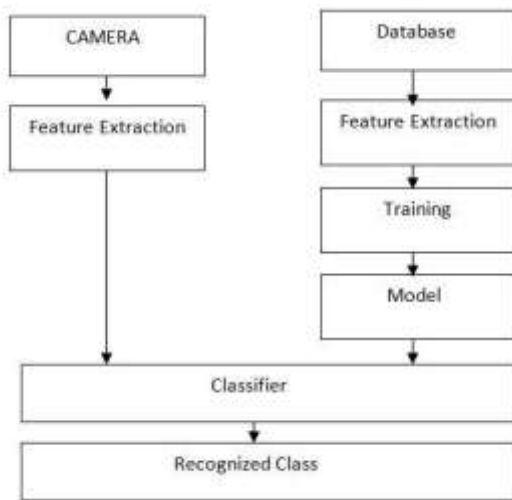


Fig. 8. System design showing camera input, database training path, feature extraction, classifier, and recognized class output.

The design emphasizes modularity. Image capture can be replaced by a mobile camera, dataset storage can be replaced by a cloud database, and the classifier can be updated without rewriting the entire system. This makes the system scalable for future improvements such as additional fruit categories, more quality grades, and integration with conveyor-belt sorting.

B. Module-Level Implementation

TABLE VI MODULE DESCRIPTION OF THE PROPOSED FRUIT QUALITY DETECTION SYSTEM

Module	Inputs	Outputs	Core Operation
Image acquisition	Camera/file image	RGB fruit image	Capture or upload image
Preprocessing	Raw image	Enhanced image	FIR filtering, normalization, background handling
Feature extraction	Enhanced image	Feature maps/vector	CNN and SPP-based feature aggregation
Feature optimization	CNN-SPP vector	Refined feature vector	SSDAE denoising and compression
Classification	Optimized feature vector	Fresh/Rotten label	SVM decision boundary
Result display	Predicted label and score	User-readable output	Display class and confidence

The image acquisition module supports multiple image formats such as JPG, PNG, and BMP. The preprocessing module standardizes the input and improves defect visibility. The feature extraction module identifies patterns related to texture, color, and surface condition. The classification module returns the final decision along with a probability or confidence score that can be displayed to the user.

The project can be implemented in Python using libraries such as TensorFlow, Keras, OpenCV, NumPy, and scikit-learn. A web-based user interface can be developed using Flask, HTML, CSS, and JavaScript, or a desktop prototype can be implemented in a notebook environment for experimentation.

C. Baseline CNN Models

The thesis evaluates three baseline models before introducing the proposed hybrid model. Manual CNN is a custom network trained from scratch. LeNet-5 is an early CNN architecture with convolution, pooling, and fully connected layers. AlexNet is deeper and

demonstrates the advantage of learning more expressive feature representations. These baselines provide a meaningful comparison for measuring the improvement achieved by the proposed CNN-SPP and SSDAE-SVM framework.

Manual CNN is relatively simple and useful for understanding convolution, pooling, activation, flattening, and dense layers. LeNet-5 is lightweight and converges quickly, but it was originally designed for small grayscale inputs. AlexNet introduces deeper convolutional layers and stronger feature learning. However, all baseline models still require fixed image sizes and may be affected by overfitting or environmental variation.

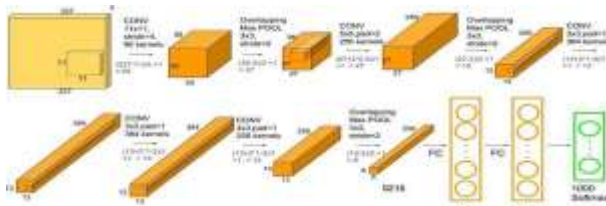


Fig. 9. AlexNet architecture used as a deeper baseline model for fruit image classification.

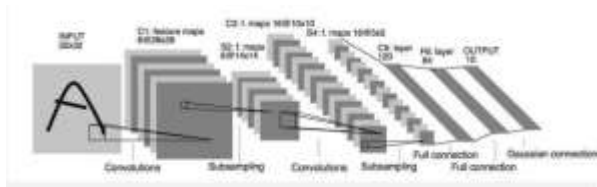


Fig. 10. LeNet-5 architecture showing convolution, subsampling, fully connected and output layers.

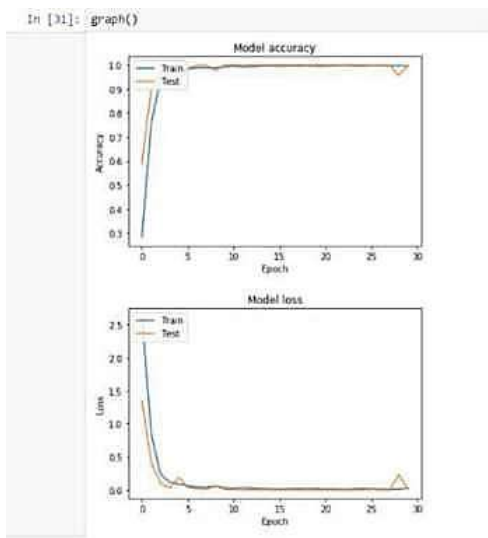


Fig. 11. Training accuracy and loss behavior for a baseline CNN model.

TABLE VII BASELINE MODEL CHARACTERISTICS

Model	Architecture Type	Strength	Limitation
Manual CNN	Custom convolutional network	Simple and easy to implement	Moderate accuracy and overfitting
LeNet-5	Classic seven-layer CNN	Fast convergence and lightweight design	Limited representation capacity
AlexNet	Deeper CNN architecture	Higher accuracy than earlier models	Requires fixed image size and more computation
Proposed hybrid	CNN-SPP + SSDAE-SVM	High accuracy and robust features	More complex training pipeline

D. Algorithmic Flow

The algorithm begins by reading the input fruit image and applying preprocessing. CNN layers generate feature maps. SPP converts these feature maps into a fixed-length vector. SSDAE refines the vector by denoising and compression. The refined features are classified by SVM, and the final output is displayed as fresh or rotten. This flow can be implemented as offline batch classification or as real-time prediction through a camera interface.

TABLE VIII PROPOSED ALGORITHM FOR FRUIT QUALITY DETECTION

Step	Operation
1	Acquire fruit image from camera or dataset
2	Apply FIR filtering, normalization and background handling
3	Extract visual features using CNN convolutional layers
4	Apply Spatial Pyramid Pooling to handle variable image sizes
5	Optimize features using Stacked Sparse Denoising Autoencoder
6	Classify optimized features using SVM
7	Display quality result as Fresh or Rotten with confidence score

V. EXPERIMENTAL SETUP

The experimental setup follows the thesis implementation environment and is suitable for M.E. laboratory work. The system can be developed using Python with deep learning and machine learning libraries. Input images are grouped by quality category and passed through baseline and proposed models. The model is evaluated using accuracy, precision, recall, F1-score, specificity, and prediction error. These metrics are important because fruit quality detection should minimize both false fresh predictions for rotten fruits and false rotten predictions for good fruits.

TABLE IX EXPERIMENTAL ENVIRONMENT AND TOOLS

Category	Specification
Programming language	Python
Deep learning framework	TensorFlow / Keras
Machine learning	scikit-learn SVM
Image processing	OpenCV and NumPy
Development environment	Jupyter Notebook / suitable IDE
Target classes	Fresh and Rotten
Performance metrics	Accuracy, precision, recall, F1-score, specificity and error rate

A. Evaluation Metrics

Accuracy measures the proportion of total images classified correctly. Precision measures how many predicted fresh or rotten images are actually correct for the target class. Recall measures how many actual class samples are recovered by the model. F1-score provides a harmonic mean of precision and recall. Specificity is especially important for medical and quality inspection style tasks because it measures how effectively the system rejects negative samples.

In a fruit grading scenario, high recall helps identify most rotten fruits, while high precision prevents unnecessary rejection of fresh fruits. Therefore, a balanced F1-score and low error rate are preferable to accuracy alone.

$$Accuracy = (TP + TN) / (TP + TN + FP + FN). \quad (3)$$

$$Precision = TP / (TP + FP), \quad Recall = TP / (TP + FN). \quad (4)$$

$$F1\text{-score} = 2 \times Precision \times Recall / (Precision + Recall). \quad (5)$$



Fig. 12. Model summary screenshot for one of the fruit classification networks.

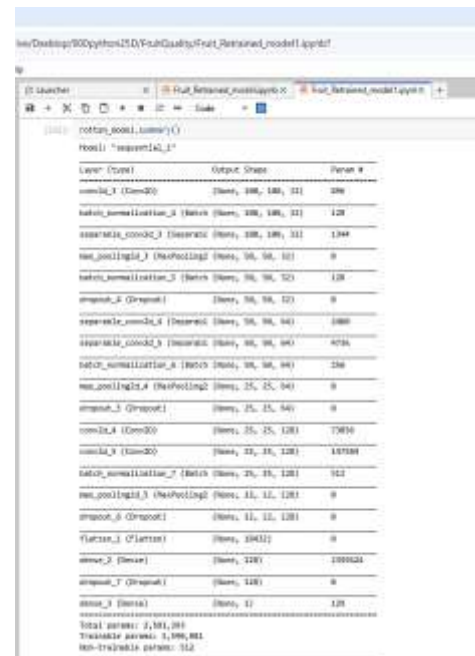


Fig. 13. Detailed model summary showing layer output shapes and trainable parameters.

VI. RESULTS AND DISCUSSION

A. Performance of Baseline Models

The initial phase evaluates Manual CNN, LeNet-5, and AlexNet. The results show that model depth and feature learning capacity improve performance. Manual CNN achieved moderate validation accuracy, LeNet-5 improved the result, and AlexNet produced the highest baseline performance. However, the fixed-size input

limitation remained a common issue for these architectures.

TABLE X PERFORMANCE SUMMARY OF PHASE I MODELS

Model	Training Accuracy (%)	Validation Accuracy (%)	Loss
Manual CNN	92.10	88.35	0.25
LeNet-5	94.25	91.45	0.18
AlexNet	97.80	95.65	0.12

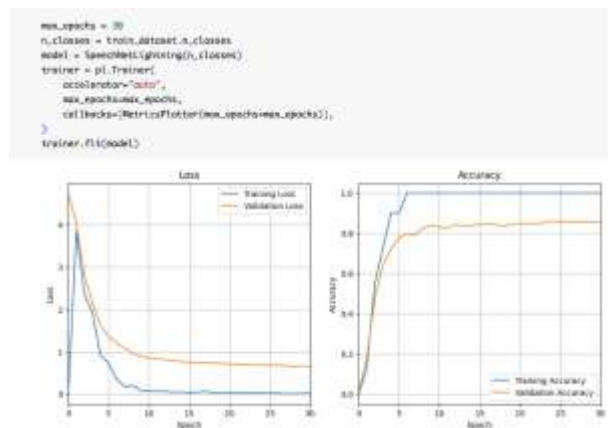


Fig. 14. Training and validation loss/accuracy graphs for a baseline model.

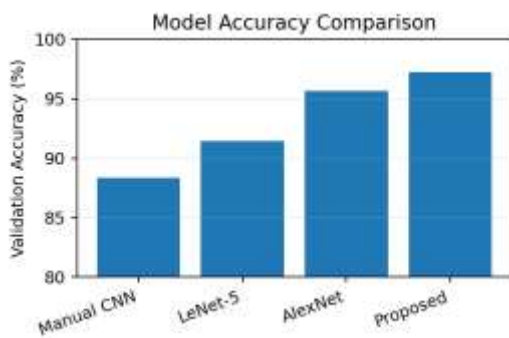


Fig. 15. Graphical comparison of validation accuracy for baseline and proposed models.

B. Performance of the Proposed Hybrid Model

The proposed CNN-SPP and SSDAE-SVM model achieved the strongest performance in the thesis. The accuracy of 97.25% indicates that the model learned discriminative visual features for distinguishing fresh and rotten fruits. Precision of 98.10% indicates that most predicted positive cases were correct. Recall of 98.98% indicates that the model detected almost all relevant samples. The F1-score of 98.81% confirms a strong balance between precision and recall.

TABLE XI PERFORMANCE METRICS OF THE PROPOSED HYBRID MODEL

Metric	Value
Accuracy	97.25%
Precision	98.10%
Recall	98.98%
F1-score	98.81%
Specificity	95.62%
Error rate	0.15

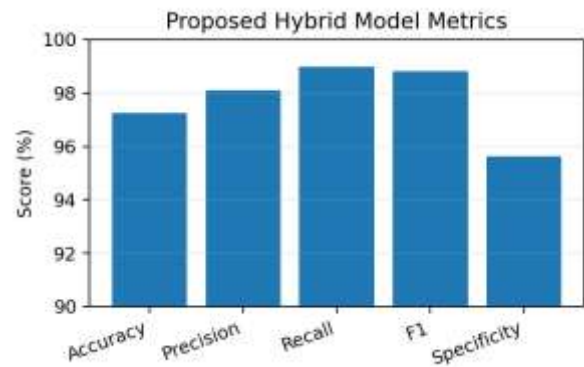


Fig. 16. Performance metric chart of the proposed CNN-SPP and SSDAE-SVM model.

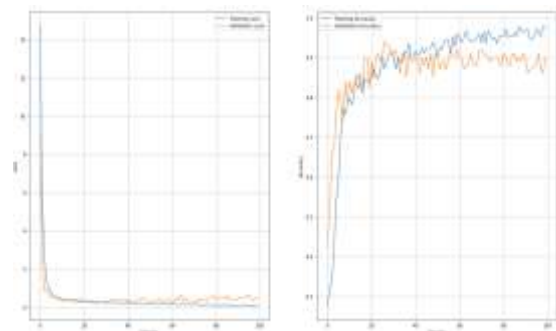


Fig. 17. Training and validation loss and accuracy curves showing model learning behavior.

C. Comparative Analysis

TABLE XII MODEL ACCURACY COMPARISON

Model	Accuracy (%)
Manual CNN	88.35
LeNet-5	91.45
AlexNet	95.65
Proposed CNN-SPP + SSDAE-SVM	97.25

The comparison indicates that accuracy improves as the model becomes more capable of learning robust visual patterns. AlexNet outperformed Manual CNN and LeNet-5 due to its deeper representation. The proposed hybrid model further improved performance by addressing key weaknesses of baseline CNNs. SPP reduced the need for resizing, SSDAE improved feature quality, and SVM produced a sharper decision boundary.

The improvement is meaningful for agricultural applications because even small increases in accuracy can reduce economic loss when applied to large-scale sorting lines. In addition, high recall is valuable because missed rotten fruits can contaminate storage batches and reduce consumer trust.

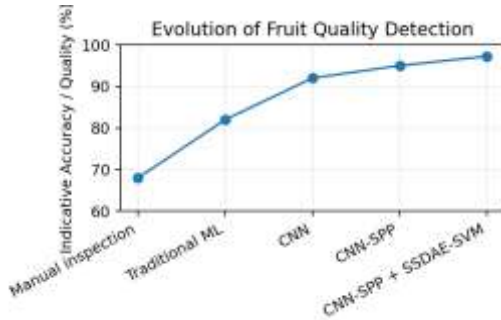


Fig. 18. Indicative evolution from manual inspection to hybrid deep learning based fruit quality detection.

D. Ablation-Based Interpretation

TABLE XIII CONTRIBUTION OF PROPOSED COMPONENTS

Component Added	Expected Improvement	Explanation
FIR preprocessing	Cleaner input features	Reduces noise while preserving important edges
CNN layers	Automatic feature extraction	Learns texture, color, and shape patterns
SPP layer	Resolution preservation	Avoids information loss from forced resizing
Dropout	Reduced overfitting	Improves validation performance
SSDAE	Robust feature representation	Suppresses noise and redundant features
SVM	Clear class separation	Improves binary decision boundary

The ablation interpretation explains why the proposed hybrid model outperforms individual baselines. The model does not rely on a single change; rather, it combines preprocessing, robust feature extraction, feature optimization, and strong classification. This type of layered improvement is appropriate for engineering research because each component can be justified and tested independently.

E. Output and Prediction Visualization

The source thesis includes screenshots of dataset organization, model summaries, training logs, training curves, and prediction outputs. These screenshots

demonstrate the practical implementation of the project and help M.E. students understand how the algorithm is executed, evaluated, and visualized in a programming environment.



Fig. 19. Training summary screenshot showing epoch-wise learning information.

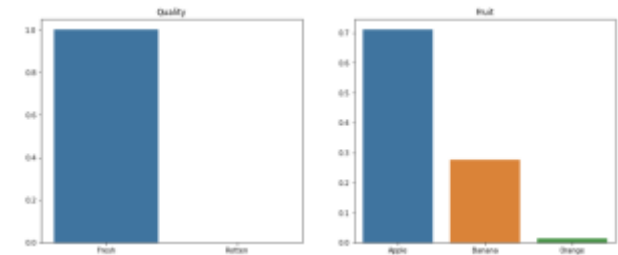


Fig. 20. Prediction visualization showing fresh quality and fruit category probabilities.

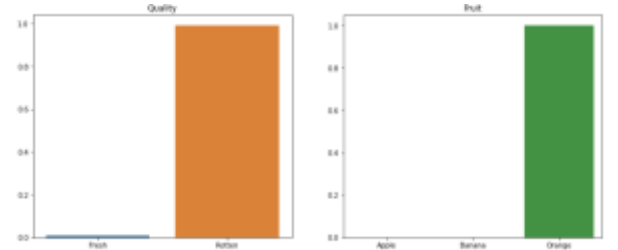


Fig. 21. Prediction visualization for rotten orange class probability.

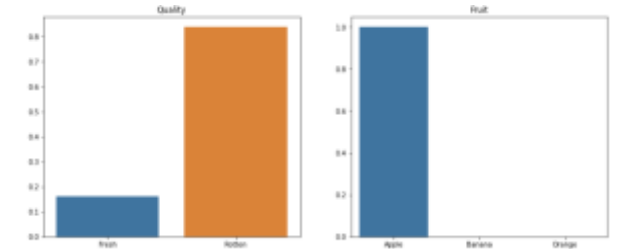


Fig. 22. Prediction visualization for rotten quality and apple category probability.

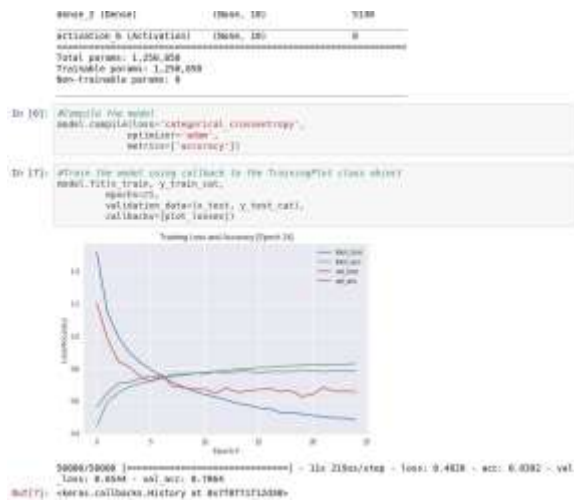


Fig. 23. Additional training graph and model compilation screenshot included in the thesis results section.

VII. CONCLUSION

This article presented fruit quality detection using deep learning techniques. The work addresses the limitations of manual inspection and conventional image processing by proposing a hybrid CNN-SPP and SSDAE-SVM framework. The system processes fruit images, enhances them through preprocessing, extracts deep visual features, preserves spatial information using SPP, optimizes features using SSDAE, and classifies quality using SVM.

The thesis results demonstrate that the proposed hybrid model achieves 97.25% accuracy, 98.10% precision, 98.98% recall, 98.81% F1-score, and 95.62% specificity. These values show that the model is effective for distinguishing fresh and rotten fruits. The hybrid approach is suitable projects because it includes image processing, deep learning, feature optimization, machine learning classification, performance evaluation, and practical system design.

The proposed system has strong potential for agricultural automation, postharvest quality control, retail inspection, and smart supply chain applications. With larger datasets, real-time deployment, and multi-class grading support, the system can be developed into a more complete fruit quality assessment platform.

REFERENCES

- [1] A. Ren, A. Zahid, A. Zoha, S. A. Shah, M. A. Imran, A. Alomainy, and Q. H. Abbasi, "Machine learning-driven approach towards the quality assessment of fresh fruits using non-invasive sensing," *IEEE Sensors Journal*, vol. 20, no. 4, pp. 2075-2083, Feb. 2020.
- [2] E. Bellocchio, T. A. Ciafuglia, G. Costante, and P. Valigi, "Weakly supervised fruit counting for yield

estimation using spatial consistency," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2348-2355, 2019.

- [3] G. Ciocca, G. Micali, and P. Napoletano, "State recognition of food images using deep features," *IEEE Access*, vol. 8, pp. 32003-32017, Feb. 2020.
- [4] G. Xuan, C. Gao, Y. Shao, M. Zhang, Y. Wang, J. Zhong, and Q. Li, "Apple detection in natural environment using deep learning algorithms," *IEEE Access*, vol. 8, pp. 216772-216780, 2020.
- [5] G. Zeng, "Fruit and vegetables classification system using image saliency and convolutional neural network," in *Proc. IEEE 3rd Information Technology and Mechatronics Engineering Conf.*, 2017.
- [6] M. Muladi, D. Lestari, and D. T. Prasetyo, "Classification of eligibility consumption of Manalagi apple fruit varieties using backpropagation," in *Proc. Int. Conf. Advanced Mechatronics, Intelligent Manufacture and Industrial Automation*, 2019.
- [7] S. Luo, K. Sarabandi, L. Tong, and L. Pierce, "A SAR image classification algorithm based on multi-feature polarimetric parameters using FOA and LS-SVM," *IEEE Access*, vol. 7, pp. 175259-175276, Dec. 2019.
- [8] W. Castro, J. Oblitas, M. De-La Torre, C. Cotrina, K. Bazan, and H. A. George, "Classification of cape gooseberry fruit according to its level of ripeness using machine learning techniques," *IEEE Latin America Transactions*, 2019.
- [9] Y. Peng, M. Liao, Y. Song, Z. Liu, H. He, H. Deng, and Y. Wang, "FB-CNN: Feature fusion-based bilinear CNN for classification of fruit fly image," *IEEE Access*, vol. 8, pp. 2169-2180, 2020.
- [10] Y.-P. Huang, T.-H. Wang, and H. Basanta, "Using fuzzy Mask R-CNN model to automatically identify tomato ripeness," *IEEE Access*, vol. 8, pp. 207672-207682, Nov. 2020.
- [11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2016, pp. 770-778.
- [12] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84-90, 2017.
- [13] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278-2324, 1998.

- [14] K. Grauman and T. Darrell, "The pyramid match kernel: Discriminative classification with sets of image features," in Proc. IEEE Int. Conf. Computer Vision, 2005, pp. 1458-1465.
- [15] C. Cortes and V. Vapnik, "Support-vector networks," Machine Learning, vol. 20, no. 3, pp. 273-297, 1995.
- [16] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," arXiv:1207.0580, 2012.