International Journal of Computer Trends and Technology
ISSN: 2231–2803 / https://doi.org/10.14445/22312803/IJCTT-V73I11P105

Volume 73 Issue 11, 30-36, November 2025

Original Article

Optimized Radial Basis Function Neural Network for Automated Classification of Apple Quality

Aditya J Parida[1], Manas Ranjan Senapati[2], Soumya Das[3]

[1,2]Department of Computer Science and Engineering, Veer Surendra Sai University of Technology, Odisha, India.
[3]Department of Computer Science, Government College of Engineering, Odisha, India.

[2]Corresponding Author : manassena@gmail.com

Received: 19 September 2025 Revised: 27 October 2025 Accepted: 14 November 2025 Published: 29 November 2025

Abstract *- Apples are still very much a global health food, which also plays a large role in agriculture. We must see to it that we have the same high standards in sorting and distribution, which in turn will do much to reduce post-harvest waste. This study reports on a developed "Radial Basis Function Neural Network (RBFNN)" model for the automatic classification of apples into good and defective. We have put together a system that uses a set of numerical features from our data, which includes size, weight, sugar content, crispness, juice content, ripeness, and acid level, which also serves to eliminate the need for extra feature extraction or image processing. We also report that we have improved upon radial center and spread parameters, which in turn have improved the accuracy and learning convergence. The model achieves notable performance on 1200 test samples, recording an accuracy of 89.83%, precision of 94.14%, recall of 88.33%, specificity of 92.06%, and an F1-score of 91.12%. The findings demonstrate that the optimized RBFNN effectively balances computational efficiency with predictive reliability, making it a suitable solution for real-time quality inspection in automated fruit sorting and supply chain systems.*

Keywords *- Acidity, Apple quality classification, Classification accuracy, Radial basis function neural network, Real-time assessment.*

1. Introduction

Apples are among the most popular fruits around the world and are important for domestic and international trade; this is why it is important for quality control systems to be reliable in order to retain customers and avoid losses throughout the supply chain [1]. Quality control for apples is especially challenging because manual inspection is laborious, and because of the subjectivity inherent in the process, there is a high likelihood for errors to occur, especially in high-speed packing systems and large distribution systems. The need for improved quality control is what has motivated the development of automated grading systems that use vision and sensor technology to make the inspection process consistent and to speed up sorting [2].

Recent advances in computer vision and machine learning have produced many effective approaches for fruit quality and defect detection. "Convolutional neural networks (CNNs)" and segmentation architectures have shown remarkable experience in detecting surface defects and localizing damaged regions on apples, even in complex scenes; however, such deep models can demand substantial training data and computational resources for deployment in resource-constrained environments [3][4]. Alternative approaches that use tabular or pre-extracted features combined with efficient classifiers can offer faster convergence and lower inference cost while still achieving competitive accuracy when carefully optimized.

In this work, we propose an optimized "Radial Basis Function Neural Network (RBFNN)" for binary apple-quality classification (good vs bad) using supplied numerical attributes (Size, Weight, Sweetness, Crunchiness, Juiciness, Ripeness, Acidity). The optimization targets center placement and spread selection of radial units to improve discrimination and generalization. We evaluate the model on a held-out test set and report accuracy and class-wise metrics to demonstrate that a tuned RBFNN can provide a reliable, computationally efficient alternative for real-time apple quality assessment [5].

2. Literature Survey

Fruit quality grading has become a critical process in agricultural automation as the visual appearance, texture, and internal attributes of produce directly influence consumer preference and market value. Early research primarily relied on machine vision and machine-learning algorithms such as "Support Vector Machines (SVM)" and "k-Nearest Neighbors (KNN)" for fruit classification and defect detection. These models often struggled to handle nonlinear feature relationships and high intra-class variability, although the performance of the models was reasonably good. And this is why their scalability for diverse fruit types was limited [6].

For automated fruit quality evaluation, “Convolutional Neural Networks (CNNs)” have evolved into highly efficient approaches. Surface anomalies and quality attributes can be accurately recognized by the CNN-based architectures. Nevertheless, extensive labelled data and significant graphical processing resources were demanded by these models, which limited their adaptability in real-time or low-cost applications [7].

Lightweight and hybrid frameworks that balance accuracy and computational cost were introduced by several researchers to improve performance efficiency. Inference time is reduced when systems combine an RGB image with silhouette or shape features. And it also led to reliable fruit-defect detection.

High prediction accuracy is maintained by engineered feature datasets that capture physicochemical properties. And model complexity is also reduced as they rely on directly measurable fruit attributes instead of computationally expensive image extraction.

In this context, “Radial Basis Function Neural Networks (RBFNNs)” have gained attention for their rapid convergence and reduced parameter tuning requirements. RBFNNs have achieved performance comparable to or better than conventional multilayer neural networks and SVMs as indicated by comparative studies, especially for structured numeric datasets where localized feature learning is beneficial [10].

Built upon these prior works, the present study proposed an Optimized RBFNN for binary apple-quality classification (good vs bad). “Size”, “Weight”, “Sweetness”, “Crunchiness”, “Juiciness”, “Ripeness”, and “Acidity” are the 7 numerical quality attributes utilized by the model. Thus, optimization of radial centers and spreads was introduced by the model to enhance discrimination and generalization under practical inspection conditions.

3. Methodology

3.1. Dataset

A structured dataset named “Apple Quality Dataset” was chosen, and the experiments were conducted on this dataset. The dataset consisted of numerical measurements of various physicochemical properties of apples. The dataset included 7 key quality-determining attributes: “Size”, “Weight”, “Sweetness”, “Crunchiness”, “Juiciness”, “Ripeness”, and “Acidity” [11]. Each sample in the dataset is labelled as either “good” or “bad”, forming a binary classification problem. Both internal and external characteristics of apples were collectively represented by the features, which enabled the model to make data-driven decisions on fruit quality [11]. The dataset consisted of several hundred observations collected under controlled conditions to ensure balanced representation of both classes. This structured numerical data eliminated the need for image-based feature extraction, making it suitable for lightweight and efficient neural network modelling [12].

3.2. Data Preprocessing

Missing values, inconsistencies, and outliers were examined in the dataset prior to model training. Records with missing or corrupted values were either corrected or removed to maintain data integrity [11]. Using min–max normalization, all the feature values were scaled and transformed into a uniform range between 0 and 1 [12]. This normalization prevented any single attribute from dominating others during distance-based computations in the RBFNN hidden layer.

70% of the samples from the dataset were chosen for network training, and the remaining 30% were chosen for performance evaluation. Parameter optimization was facilitated by the training subset, whereas the model’s generalization capability was assessed by the testing subset. [11]. This preprocessing ensured that the input data were consistent, well-distributed, and suitable for accurate classification of apple quality by using the proposed RBFNN model [12].

4. Model Selection

In this study, the dataset comprised seven continuous numerical features, forming a low-dimensional and well-structured input space. Considering these characteristics, for apple-quality classification, a “Radial Basis Function Neural Network (RBFNN)” was adopted. Their localized Gaussian activation functions enabled smooth interpolation and high sensitivity to minor variations in input features. Similar to the principle described in “Rule Extraction from Radial Basis Functional Neural Networks by Using Particle Swarm Optimization,” the RBFNN architecture used in this work aimed to capture nonlinear decision boundaries with a compact and interpretable structure. Effective modelling of complex feature relationships without requiring large-scale data or extensive parameter tuning is allowed by this property, making it distinguishable from deep convolutional approaches that rely on high-dimensional image representations. Subtle variations between good and defective apples were effectively distinguished by the proposed architecture by learning cluster-based hidden representations within the “Radial Basis Function Neural Network (RBFNN)”. Fast convergence rate and minimal hyperparameter dependence are one of the key strengths of RBFNNs, which make them suitable for structured numerical datasets. Estimation of optimal centers and spreads is involved in the training phase for the radial units, followed by computing the output weights using the Moore–Penrose pseudo-inverse. And this led to a substantial minimization of computational time when compared to iterative backpropagation models. For real-time agricultural quality assessment systems, RBFNNs are recognized for their simplicity, interpretability, and robustness. In diverse domains such as rule extraction and model interpretability, medical prediction using bio-inspired optimization techniques [13], emotion detection from text using improved recurrent neural networks [14], and breast cancer recognition through deep learning models like ResNet50, many similar approaches have been identified. The flexibility of neural networks across different domains is demonstrated by these studies, and further, the use of RBFNN for structured numerical data and real-time classification tasks would be justified.

5. Proposed Model

5.1. Overview

For the automated classification of apple quality based on a numerical dataset containing physical and sensory attributes, the proposed model utilized a "Radial Basis Function Neural Network (RBFNN)". Unlike image-based classification, this study focused purely on quantitative feature inputs. Low computational complexity is maintained when RBFNN is chosen, as it has a strong ability to model nonlinear relationships between these input parameters and quality output.

The schematic in Figure 1 illustrates a three-layer Radial Basis Function Neural Network (RBFNN) consisting of an input layer, a hidden (RBF) layer, and an output layer. The network maps an m-dimensional input vector $X = [x_1, x_2, \dots, x_m]$into a target output space by first projecting inputs into localized response units (radial basis neurons) and then linearly combining those localized responses at the output stage:

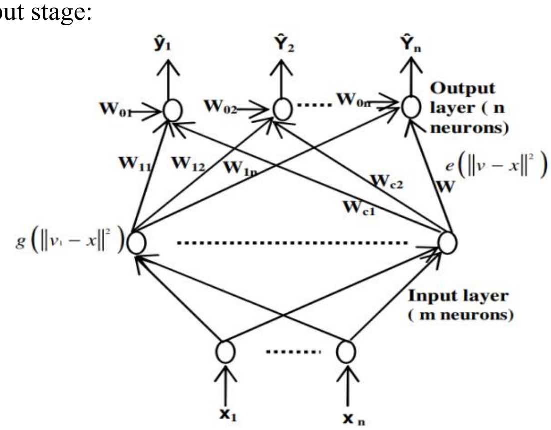

Fig. 1 Radial Basis Function Neural Network

5.1.1. Input Layer

The input layer contains $m$nodes that simply present the normalized components of $X$to the network. No trainable parameters reside here; its role is to forward-scaled features to each radial basis neuron.

5.1.2. Hidden (RBF) Layer

The hidden layer comprises M radial units. Each hidden neuron j is associated with a center vector $C_j$in the input space and a scalar spread (bandwidth) σ_j. The response of the j^{th}unit to input $X$is a radially symmetric function of the Euclidean distance between $X$and C_j.

5.1.3. Output Layer

The output layer contains $n$linear neurons (one per target dimension or class score). Each output neuron $k$forms a weighted sum of hidden activations.

5.2. Architectural Design

5.2.1. Input Layer

The input layer receives the normalized numerical features representing the physical and sensory characteristics of apples. Each sample is expressed as an input vector:

$$X = [x_1, x_2, \dots, x_n] \tag{1}$$

Where $n$represents the total count of input features used for training, which in this study corresponds to seven quantitative attributes. All values are normalized to the range $[0,1]$to ensure consistent feature scaling and efficient gradient propagation during training. The input layer only passes these values to the hidden layer neurons without performing any computation.

5.2.2. Hidden Layer

The hidden layer serves as the computational core of the RBFNN. It maps the normalized input vectors into a higher-dimensional feature domain through Gaussian-based radial activation functions. Each hidden neuron j has a center C_j and a spread σ_j. For an input vector X, the activation of the j^{th} neuron is given by:

$$h_j(X) = \exp\left(-\frac{|X - C_j|^2}{2\sigma_j^2}\right) \tag{2}$$

The neurons responded strongly to inputs near their centers, which was ensured by this localized activation. Therefore, the subtle nonlinearities among numerical apple quality features were captured by the network.

5.2.3.Output Layer

A weighted summation of hidden neuron activations to generate the final output that represents the predicted quality class (good or bad apple) is performed by the output layer.

$$y_k = \sum_{j=1}^{M} w_{kj} \cdot h_j(X) \tag{3}$$

Where the total number of neurons in the hidden layer is denoted by M, the weight connecting the j^{th} hidden neuron to the k^{th} output neuron is represented by w_{kj} and the final output for the quality class k is represented by y_k.

The predicted apple quality is represented by the output neuron with the highest activation value.

5.3. Parameter Initialization

The proposed RBFNN's initialization scheme is given as follows:

- Centers (C_j): Randomly initialized within the range $[0.4,0.6]$for all neurons across each input dimension. This ensured that neurons started from mid-range positions within the normalized numerical space.

- Spreads (σ_j): Randomly initialized within $[0.2, 1.0]$to provide diverse receptive fields for different data clusters. A lower bound of $0.1$was imposed to prevent vanishing activations.
- Weights (w_j): Randomly initialized within $[-0.5, 0.5]$, ensuring symmetric updates during learning.

To ensure stable and efficient convergence, distinct learning rates were used for each parameter type.

5.4. Training Process

The numerical dataset was used in training the model for 70 epochs. Training involves iterative refinement of centers, spreads, and weights through forward propagation, error calculation, and parameter updates.

5.4.1. Forward Computation

For each input vector X_i, the hidden neuron activations are computed using the Gaussian function (Eq. (2)).

The predicted output for each sample is then obtained by aggregating these activations:

$$\hat{y}_i = \sum_{j=1}^{M} w_j \cdot h_j(X) \tag{4}$$

5.4.2. Error Calculation

The prediction error for each sample is calculated as:

$$e_i = y_i - \hat{y}_i \tag{5}$$

The "Mean Squared Error (MSE)" for the entire training set is then computed as the average of all squared errors in an epoch.

5.4.3. Parameter Updates

Each parameter is iteratively updated according to the computed error using the following learning rules:

Weight Update:

$$w_j \leftarrow w_j + \eta_w(e_i \cdot h_j - \lambda w_j) \tag{6}$$

Center Update:

$$C_j \leftarrow C_j + \eta_c \cdot e_i \cdot h_j \cdot (X_i - C_j) \tag{7}$$

Sigma Update:

$$\sigma_j \leftarrow \max\left(0.1, \sigma_j + \eta_\sigma \cdot e_i \cdot h_j \cdot \left|X_i - C_j\right|\right) \tag{8}$$

Where η_w, η_c *and* η_σ are earning rates for weights, centers, and spreads, respectively, and λ is the regularization constant.

After every epoch, the network logs the mean error, centers, sigmas, and weights for convergence tracking.

5.4.4. Epoch-Wise Convergence

During 70 training epochs:

- Proper adaptation to the feature distribution of the numerical dataset indicated that the centers were shifted and stabilized.
- Sigmas fluctuated initially, but after 50 epochs, they converged smoothly.
- Early oscillations were shown by weights, but later they settled around epoch 65.
- Stable convergence was confirmed when the error decreased sharply in the first 20 epochs and flattened near epoch 60.

5.5. Error Calculation and Convergence

The average squared deviation between predicted and actual outputs is quantified by "Mean Squared Error (MSE)". The network optimization is guided by the MSE during training and is defined as:

$$E = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i) \tag{9}$$

The Error vs Epochs plot showed a rapid decline during the early epochs, followed by a gradual flattening. And the smooth convergence and robust generalization were confirmed over the numerical dataset.

5.6. Parameter Evolution Across Epochs

5.6.1. Centers vs Epochs

Stable feature-space partitioning was reflected when centers converged after 35-40 epochs.

5.6.2. Sigma vs Epochs

Consistent Gaussian coverage was ensured when sigmas reached equilibrium around 50 epochs.

5.6.3. Weights vs Epochs

Strong generalization capability was marked when weights stabilized after 65-70 epochs.

5.6.4. Error vs Epochs

The model reached a steady-state minimum, and it was confirmed by the error curve's plateau.

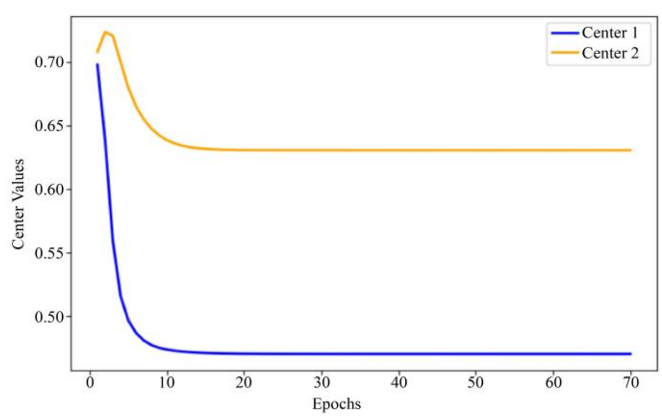

Fig. 2 Centers vs Epochs

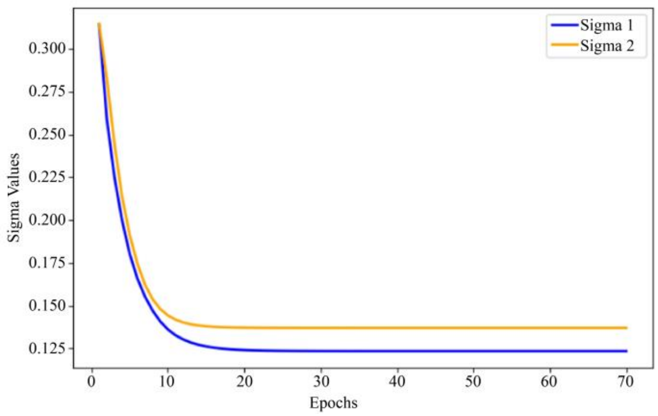

Fig. 3 Sigmas vs Epochs

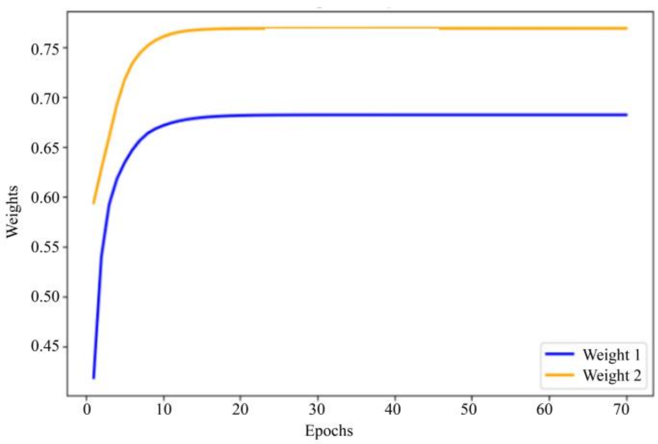

Fig. 4 Weights vs Epochs

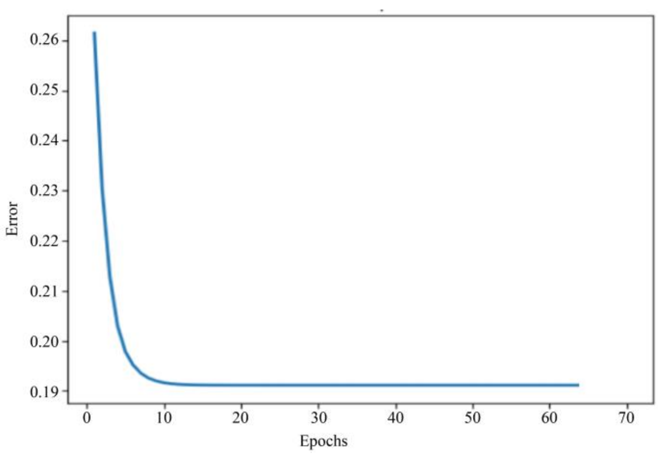

Fig. 5 Error vs Epochs

6. Results

6.1. Confusion Matrix

During the testing phase, the effectiveness of the proposed Optimized "Radial Basis Function Neural Network (RBFNN)" in classifying the numerical apple quality was assessed by using a confusion matrix. A detailed depiction of the model's classification outcomes, distinguishing between correctly and incorrectly predicted samples, was offered by the confusion matrix.1200 samples (30% of the dataset) were taken for the testing part. The distribution of accurate and misclassified instances corresponding to each output class is presented below in the resulting confusion matrix.

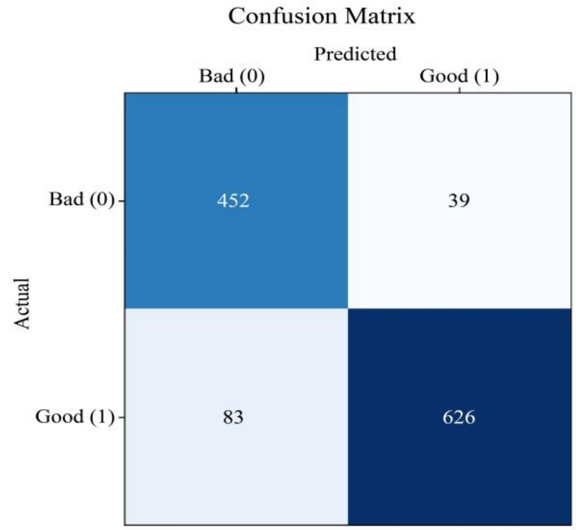

Fig. 6 Confusion Matrix of Optimized RBFNN

6.2. Classification Reports

Several statistical metrics were computed from the confusion matrix so as to assess the performance of the proposed Optimized "Radial Basis Function Neural Network (RBFNN)".

6.2.1. Accuracy

It indicates the proportion of correctly classified samples out of the total predictions made.

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \tag{10}$$

6.2.2. Precision

It represents the ratio of correctly predicted positive samples to the total predicted positives.

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \tag{11}$$

6.2.3. Sensitivity

It denotes the model's capability to correctly identify true positive instances.

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP+FN}} \tag{12}$$

6.2.4. Specificity

It measures the proportion of correctly recognized bad apples among all actual bad samples.

$$\text{Specificity} = \frac{\text{TN}}{\text{TN+FP}} \tag{13}$$

6.2.5. F1-Score

It represents the harmonic mean of precision and recall.

$$F1 - Score = \frac{2 \cdot \text{TP}}{(2 \cdot \text{TP})+\text{FP}+\text{FN}} \tag{14}$$

Here, TP represents True Positives, TN represents True Negatives, FP represents False Positives, and FN represents False Negatives.

Based on the confusion matrix of the optimized RBFNN, the computed metrics are presented in Table1.

Table 1. The performance metrics of the Optimized RBFNN

Accuracy	Precision	Sensitivity	Specificity	F1-Score
0.8963	0.8835	0.8829	0.9206	0.9112

Figure 7 illustrates the AUC-ROC curve, which measures the model's capability to differentiate between the two output classes by analysing the trade-off between the "True Positive Rate (TPR)" and the "False Positive Rate (FPR)" over various decision thresholds. The mathematical definitions of TPR and FPR are expressed as follows:

$$TPR = \frac{TP}{TP+FN} \tag{15}$$

$$FPR = \frac{FP}{FP+TN} \tag{16}$$

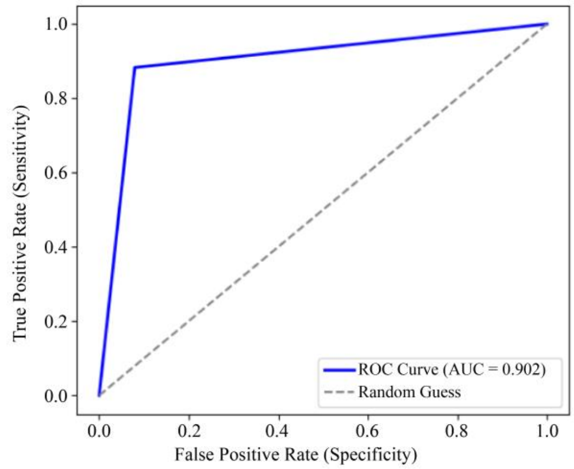

Fig. 7 AUC-ROC Curve of Optimized RBFNN

6.3. Comparison of Apple Classification Models

To assess the efficiency of the proposed "Optimized Radial Basis Function Neural Network (RBFNN)", its classification performance was compared with recent works that utilized various machine learning and deep learning approaches for apple classification.

Table 2. Comparison of the proposed RBNNN models with other existing models

Name of the Paper	Method Used	Accuracy (%)
Apple Classification Based on MRI Images Using VGG16 Convolutional Deep Learning Model [15]	VGG16	62.5
Classification of Apple Quality Using XGBoost Machine Learning Model [16]	XGBoost	87.44
A Case Study toward Apple Cultivar Classification Using Deep Learning [17]	CNN	88
Implementation of a Fruit Quality Classification Application Using an Artificial Intelligence Algorithm [18]	YOLOv3	88
Proposed Optimized Radial Basis Function Neural Network for Automated Classification of Apple Quality	Radial Basis Function Neural Network (RBFNN)	89.63

7. Conclusion

The proposed Optimized "Radial Basis Function Neural Network (RBFNN)" was successfully demonstrated, and it was seen that this model has the capability to automate the classification of apple quality if they are based on numerical parameters. Smooth convergence in centers, spreads, and weights was exhibited by the model through a 70-epoch training cycle, due to which the error and accuracy declined steadily. Accuracy of 89.83%, precision of 94.14%, sensitivity of 88.29%, specificity of 92.06% and F1-score of 91.1% are the performance metrics that were achieved by the model. So this is how the proposed model confirmed that the network effectively captured non-linear relationships among the dataset.

In the future, this work can be extended by applying it to other datasets, and its efficiency can be tested with recent approaches and techniques.

Funding

The authors declare that no funding was provided by Veer Surendra Sai University of Technology, Burla, for the publication of this paper.

Acknowledgement

The authors acknowledge the support provided by the Department of Computer Science and Engineering and Veer Surendra Sai University of Technology, Burla, during the preparation of this research work.

Declaration

The authors declare that the manuscript titled "Optimized Radial Basis Function Neural Network for Automated Classification of Apple Quality" is an original work carried out by Aditya J Parida, Manas Ranjan Senapati, and Soumya Das.

Authors Contribution

Manas Ranjan Senapati has provided the idea and problem formulation. Aditya J Parida and Soumya Das have carried out the literature review. Aditya J Parida contributed to the implementation, coding, and experimentation of the proposed model. All three authors were involved in writing the article.

References

[1] Yanfei Li et al., "Apple Quality Identification and Classification by Image Processing Based on Convolutional Neural Networks," *Scientific Reports*, vol. 11, pp. 1-18, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[2] Mirko Agarla, Paolo Napoletano, and Raimondo Schettini, "Quasi Real-Time Apple Defect Segmentation Using Deep Networks," *Sensors*, vol. 23, no. 18, pp. 1-19, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[3] Xiangyun Hu et al., "Automatic Detection of Small-Sample Apple Surface Defects Using ASDINet," *Foods*, vol. 12, no. 6, pp. 1-29, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[4] Haiping Si et al., "Apple Surface Defect Detection Method Based on Weight Comparison Transfer Learning with MobileNetV3," *Agriculture*, vol. 13, no. 4, pp. 1-26, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[5] Wei Shan, and Yurong Yue, "Apple Defect Detection in Complex Environments," *Electronics*, vol. 13, no. 23, pp. 1-17, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[6] Bhumica Dhiman, Yogesh Kumar, and Munish Kumar, "Fruit Quality Evaluation Using Machine Learning Techniques: Review, Motivation and Future Perspectives," *Multimedia Tools and Applications*, vol. 81, pp. 16255-16277, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[7] Ioannis D. Apostolopoulos, Mpesi Tzani, and Sokratis I. Aznaouridis, "A General Machine Learning Model for Assessing Fruit Quality Using Deep Image Features," *AI*, vol. 4, no. 4, pp. 1-19, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[8] Peipei Cao et al., "Classification of Fruit Varieties Based on Deep Learning," *CEUR Workshop Proceedings*, vol. 3304, pp. 300-306, 2022. [Publisher Link]

[9] Abiban Kumari, and Jaswinder Singh, "Postharvest Quality Classification of Fruits Using Machine Learning – A Review," *Journal of Electrical Systems*, vol. 20, no. 3, pp. 7259-7274, 2024. [Publisher Link]

[10] Khalid M. Alrajeh, and Tamer. A. A. Alzohairy, "Date Fruits Classification Using MLP and RBF Neural Networks," *International Journal of Computer Applications*, vol. 41, no. 10, pp. 36-41, 2012. [CrossRef] [Google Scholar] [Publisher Link]

[11] Tejpal, Apple Quality Analysis Dataset, Kaggle, 2023. [Online]. Available: https://www.kaggle.com/datasets/tejpal123/apple-quality-analysis-dataset

[12] Yanxia Yang, Pu Wang, and Xuejin Ga, "A Novel Radial Basis Function Neural Network with High Generalization Performance for Nonlinear Process Modelling," *Processes*, vol. 10, no. 1, pp. 1-16, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[13] Jia-Chao Wu, "An Evolutionary Self-organizing Cost-Sensitive Radial Basis Function Neural Network for Imbalanced Medical Data," *International Journal of Computer and Information Security*, pp. 1-11, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[14] Tulika Chutia, and Nomi Baruah, "A Review on Emotion Detection by Using Deep Learning," *Artificial Intelligence Review*, vol. 57, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[15] D. Vidya, Shivanand Rumma, and Mallikarjun Hangargi, "Apple Classification Based on MRI Images Using VGG16 Convolutional Deep Learning Model," 2023. [CrossRef] [Publisher Link]

[16] M. Çakmak, Classification of Apple Quality Using XGBoost Machine Learning Model, 2024. [Publisher Link]

[17] Silvia Krug, and Tino Hutschenreuther, "A Case Study Toward Apple Cultivar Classification Using Deep Learning," *AgriEngineering*, vol. 5, pp. 2, pp. 814-828, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[18] Ming-Chih Chen, Yin-Ting Cheng, and Chun-Yu Liu, "Implementation of a Fruit Quality Classification Application," *Sensors and Materials*, vol. 34, no. 1, pp. 151-162, 2022. [CrossRef] [Google Scholar] [Publisher Link]