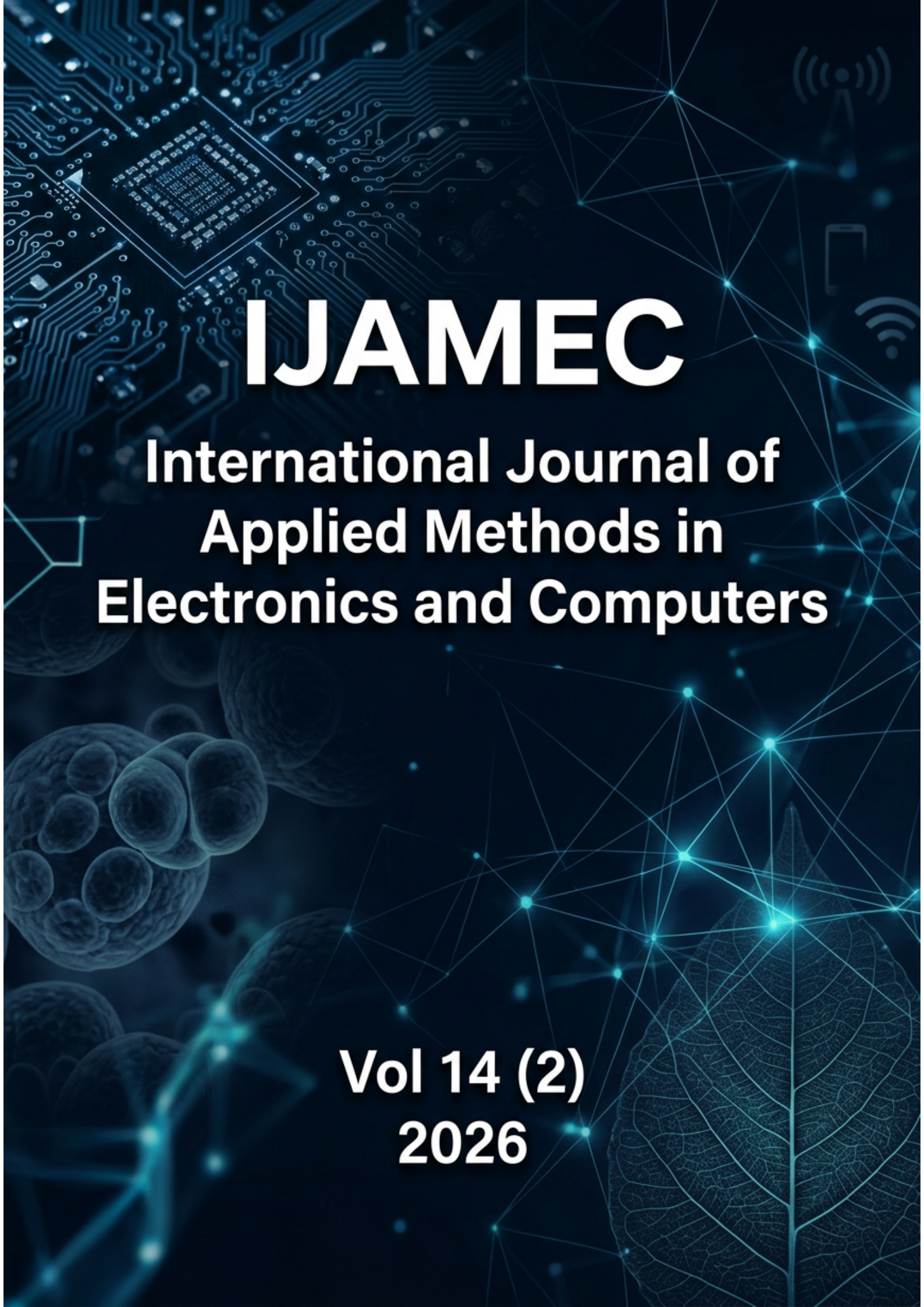


The background is a dark blue gradient featuring a complex network of glowing blue lines and nodes, resembling a digital or biological structure. In the top left, there is a detailed circuit board with a central chip. In the bottom left, there are several spherical, organic-looking structures. In the bottom right, there is a large, detailed leaf. The overall theme is a blend of technology and nature.

IJAMEC

**International Journal of
Applied Methods in
Electronics and Computers**

**Vol 14 (2)
2026**

Contents	Pages
<i>Research Articles</i>	
Analysis of Potato Diseases Using Image Processing Methods for Detection Muhammed Ali Akçay, Ibrahim Demirci, Yavuz Selim Taspinar	59-69
Performance Analysis of Machine Learning Algorithms on Wisconsin Diagnostic Breast Cancer Data Set Enriched with Data Augmentation Technique Ayça Acet, Abdullah Erhan Akkaya	70-80
An Offline-First Mobile Reporting System for Digital One Health Surveillance in Resource-Constrained Settings Edoghogho Olaye, Daniel Obuh	81-88
Comparative Evaluation of Machine Learning Algorithms for Raisin Variety Classification Based on Morphological Features Hüseyin Bulduk, Kadir SABANCI	89-101

*Research Article***Analysis of Potato Diseases Using Image Processing Methods for Detection****Muhammed Ali Akcay^{a,*} , Ibrahim Demirci^b , Yavuz Selim Taspinar^c** ^a Graduate School of Natural and Applied Sciences, Selcuk University, Konya, Türkiye^b Department of Mechatronic Engineering, Faculty of Technology, Selcuk University, Konya, Türkiye^c Department of Mechatronic Engineering, Faculty of Technology, Selcuk University, Konya, Türkiye

ARTICLE INFO

Article history:

Received 26 December 2025

Accepted 26 June 2026

*Keywords:*Potato diseases,
Image processing,
Machine learning,
Performance analysis

ABSTRACT

Potatoes are one of the basic food sources consumed by living organisms. Potatoes are naturally vulnerable to diseases. Potatoes are a food that can spoil over time, and their nutritional value gradually decreases during this process. Therefore, the quality of potatoes needs to be continuously monitored during the production phase. For this reason, image processing tools are now being used to detect potato diseases. Based on this motivation, this study uses data from 451 different potato images. This dataset contains 7 disease classes: Black Scurf, Blackleg, Common Scab, Dry Rot, Healthy Potatoes, Miscellaneous, and Pink Rot. Artificial Neural Network (ANN), K-Nearest Neighbor (KNN), Support Vector Machine (SVM), and Logistic Regression (LR) methods were used for the classification of this data. Confusion matrix and accuracy, precision, recall, and F1 Score metrics were used to analyze the classification success of the models. As a result of training and testing the models, classification success rates of 68.5% were achieved with the ANN model, 54.8% with the KNN model, 65.6% with the SVM model, and 67.0% with the LR model. The highest classification success was obtained from the ANN model. In conclusion, it can be said that all classification models can be used to detect potato diseases.

This is an open access article under the CC BY-SA 4.0 license.
(<https://creativecommons.org/licenses/by-sa/4.0/>)

1. Introduction

Advances in technology have significantly shaped agricultural policy. Particularly with the widespread use of sensor-based data collection and automation systems, the detection and quality control of potato diseases have become more systematically monitored. This is of great importance both for increasing production efficiency and for protecting the harvested potatoes. Today, many farms use smart seeding systems, and these systems provide continuous data on the health status of the plants. While these data are analyzed in conjunction with components such as emerging early changes or yield estimations, the identification of these components has also become increasingly important. Potatoes are a staple food in human nutrition, ideal for their fat, protein, and mineral content. However, these values can vary depending on irrigation methods and the minerals and systems used in

storage. Therefore, regular monitoring of potato production is necessary for both the sustainability and health of production. This capability is achieved through the use of data analysis and machine learning methods; visual data obtained from potato samples makes it possible to visualize their quality. In recent years, the use of machine learning-based models has significantly increased accuracy rates in food distribution quality control and expansion. These methods minimize human-induced evaluation errors by automatically extracting meaningful features from the obtained data, making the analysis process faster and more reliable. Many studies have been conducted in the literature regarding the identification of potato diseases. Some of these sections are listed below.

In one study, Arshaghi et al. used a data archive of 5000 images from the CFIA and USDA databases to detect diseases in potatoes using the Convolution Neural Network (CNN) method. They achieved 100%

* Corresponding author. E-mail address: akcaymuhammedali10@gmail.com
DOI: 10.58190/ijamec.2026.170

classification success using the CNN method. They contributed to the literature by demonstrating that potato diseases can be identified using a different image processing method [1].

In one study, Singh and Kaur used Plant Village Dataset data to identify diseases on potato leaves using the K-Means Methodology. They achieved an accuracy rate of 95.99% using the K-Means Methodology. Their work contributed to the literature on the detection of diseases on potato leaves [2].

In another study, Oishi et al. used data from images taken in Sapporo and Obihiro regions of Japan in June-July 2019 and 2020 to identify healthy or diseased potatoes using the Variational Autoencoder (VAE) method. They achieved an accuracy rate of 96.7% with the VAE method. Their work contributed to the literature on identifying potato diseases using portable cameras with the VAE method [3].

In yet another study, Alzakari et al. used Potato Leaf Disease Based on Weather Details data to identify diseases on potato leaves using Convolutional Neural Network-Long Short-Term Memory (CNN-LSTM) methods. They achieved 97.1% success rates with CNN-LSTM methods. They contributed to the literature by using CNN-LSTM methods in image processing and agricultural activities [4].

In one study, Zhang et al. used data from their own experiments to investigate early intervention for early blight disease in potato leaves using CNN and Multiple Scattering Correction (CNN-MS), CNN and Wavelet Transform (CNN-WT), CNN and First-Order Difference (CNN-D1), and CNN and Second-Order Difference (CNN-D2) methods. They achieved 100% success rates with CNN-WT and CNN-D2 methods. They contributed to the literature by using hyperspectral imaging technology to detect crop pests and prevent the spread of this disease [5].

In one study, Kumar and Patel used the Hierarchical Deep Learning Convolutional Neural Network (HDLCNN) method to first extract leaf characteristics, then identify leaf diseases, and finally perform proactive interventions on diseased plants using data from the Kaggle The Plant Village Dataset. The HDLCNN method yielded 16% more successful results than other methods. They have contributed to the literature by using the HDLCNN method for disease detection and categorization [6].

In one study, Anim-Ayeko et al. developed a model using The Plant Village Dataset data with the ResNet-9 method, considering leaf shape, existing diseases, and the overall green area of the leaf... They achieved an overall success rate of 99.33% with the ResNet-9 method. They contributed to the literature by providing image analysis and image processing using ResNet-9 [7].

In another study, Baron et al. used data consisting of

photographs they had taken themselves to identify plant diseases using Support Vector Machine (SVM) and CNN methods, and compared these two methods. They achieved a success rate of 97% with the CNN method and 87% with the SVM method. They contributed to the literature by enabling farmers to produce more productive crops and detect diseased crops early using these methods [8].

In yet another study, Tarik et al. Using data from The Plant Village Dataset, they performed a procedure to identify diseases in potato leaves using the CNN method. They achieved a 99% success rate with the CNN method. They have contributed to the literature by using CNN methods to detect diseases and animal attacks on potato leaves [9].

In one study, Bienkowski et al. used their own data to detect late blight and black leg diseases on potatoes using Partial Least Squares (PLS) and Backpropagation Neural Network (BPNN) methods. They achieved a 76% success rate with PLS and BPNN methods. They have contributed to the literature by using PLS and BPNN methods for early diagnosis of plant diseases through image processing [10].

In one study, Kadam et al. (2022) performed automated disease detection using machine learning and image processing techniques with 2034 potato leaf images. They achieved results with 94% accuracy using support vector machines and decision tree methods. They developed a mobile-based diagnostic device. It has been used in the field of rapid field diagnostic systems. They have contributed to the literature by providing mobile solutions that enable farmers to diagnose diseases instantly in the field [11].

In one study, Iqbal and Talukder (2020) performed automatic diagnosis of potato leaf diseases using image segmentation and machine learning with 450 images from the PlantVillage dataset. They achieved an accuracy rate of 97% using the Random Forest algorithm method. They developed an automatic classification software. It has been used in the field of artificial intelligence-supported decision systems in agriculture. They have contributed to the literature by reducing yield loss through early diagnosis of diseases [12].

In another study, Islam et al. (2017) performed disease classification using image segmentation and a multi-class support vector machine with 300 potato leaf images. They achieved an accuracy rate of 95% using the SVM algorithm method. They developed a plant health monitoring system. It has been used in the field of food safety and sustainable agriculture. They have contributed to the literature by increasing the accuracy of agricultural diagnostic systems through image processing [13].

In yet another study, Pinki et al. performed disease diagnosis using rice leaf disease data with K-means clustering and SVM methods. They achieved a great success rate of 92.06% accuracy using the SVM method. They developed a visual content-based diagnostic device.

It has been used in methods such as diagnosing cereal diseases. They have contributed to the literature with systems based on visual feature extraction [14].

In one study, Faria et al. performed image processing and classification operations with Hybrid Deep Learning methods using potato disease data consisting of 451 images verified by the Bangladesh Agricultural Research Institute. They achieved a 97% accuracy rate with the Random Forest algorithm, one of the machine learning methods. They developed an automated disease diagnosis device. They made contributions to the literature such as AI-assisted disease diagnosis in digital agriculture applications [15].

According to research in the literature, the contributions of this study can be listed as follows:

- In the study, features with high representational power were extracted from images of potato diseases with low computational cost using SqueezeNet, one of the deep learning architectures. This can contribute to the development of efficient agricultural systems, especially in environments where financial resources are limited and sufficient equipment is lacking.

- Artificial Neural Networks (ANN), K-Nearest Neighbor (KNN), Support Vector Machines (SVM), and Logistic Regression (LR) algorithms were applied to the obtained features, and the performance, accuracy, and generalization capabilities of different classification approaches were systematically compared. This comparison reveals the potential of integrating classical machine learning methods with deep features.

- In this study, multi-class diagnosis performance was evaluated using a seven-class potato disease dataset consisting of 451 images. In this respect, it goes beyond the generally two-class (diseased/normal) models in the literature and demonstrates the effectiveness of deep features in more complex classification scenarios.

- The proposed method, using a combination of transfer learning and classical machine learning models, has achieved both high accuracy and interpretability. This integrated approach offers a viable and rapid alternative for detecting diseased potatoes in potato production fields.

The rest of the article is structured as follows: Section 2, Materials and Methods, presents the dataset used in the study, the methods employed, and performance metrics. Section 3 includes the experimental results and discussions from the study. Section 4 presents the study's conclusions, contributions, limitations, and application areas.

2. MATERIAL AND METHODS

2.1. Potato Diseases Datasets (PDD)

The dataset used in the study contains a total of 451 images in 7 classes. The classes are Black Scurf, Blackleg, Common Scab, Dry Rot, Healthy Potatoes, Miscellaneous, and Pink Rot. There are 58 images in the Black Scurf class,

60 in the Blackleg class, 62 in the Common Scab class, 60 in the Dry Rot class, 80 in the Healthy Potatoes class, 74 in the Miscellaneous class, and 57 in the Pink Rot class [15, 16].

Images in the Healthy Potatoes and Miscellaneous categories are 224x224 pixels in size.

The images in the Black Scurf class consist of 37 photographs with a resolution of 224x224 pixels, as well as images with dimensions of 650x497, 244x215, 900x507, 645x376, 647x398, 485x417, 171x209, 516x597, 322x157, 500x500, 206x211, 210x210, 224x256, 265x238, 225x224, 294x286, 223x271, 205x311, 250x227, and 225x225 pixels.

The images in the Blackleg class consist of 40 photographs with a resolution of 224x224 pixels, as well as images with dimensions of 468x315, 297x169, 394x450, 138x269, 650x679, 335x262, 303x202, 250x202, 400x279, 900x597, 338x253, 304x252, 481x637, 335x254, 335x253, 255x313, 1021x680, 1770x1374, 1556x1326, and 100x838 pixels.

The images in the Common Scab class consist of 47 photographs with a resolution of 224x224 pixels, as well as images with dimensions of 318x159, 183x137, 183x123, 650x457, 182x137, 280x436, 183x137, 640x480, 547x343, 300x194, 646x450, 240x200, 685x532, and 1024x674 pixels.

The images in the Dry Rot class consist of 45 photographs with a resolution of 224x224 pixels, as well as images with dimensions of 548x365, 517x327, 238x318, 427x320, 257x196, 327x262, 640x447, 220x318, 224x318, 283x178, 600x472, 426x319, 452x360, 369x365, and 1068x580 pixels.

The images in the Pink Rot class consist of 46 photographs with a resolution of 224x224 pixels, as well as images with dimensions of 896x502, 416x121, 250x251, 704x526, 350x116, 700x842, 349x220, 396x619, 156x163, 208x242, and 1181x894 pixels.

Figure 1 shows sample images from the dataset.

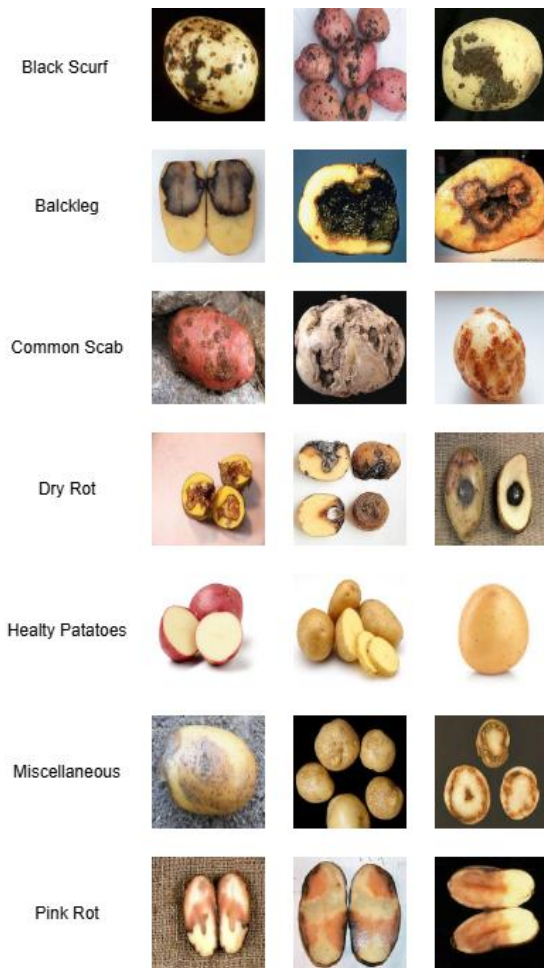


Figure 1. Shows Sample Images From The Dataset.

2.2. Convolutional Neural Networks (CNN)

Convolutional Neural Networks are one of the most important architectures of deep learning and are used especially in the processing of visual data. CNNs automatically extract features from images through filters. While simple patterns such as edges and corners are learned in the first layers, more complex structures are recognized in the deeper layers [17]. The features extracted at the end of this process are used for classification or prediction. The structure of a CNN consists of 3 layers: convolution layers, activation functions, pooling layers, and fully connected layers. Convolution layers generate feature maps, activation functions such as ReLU learn nonlinear relationships, pooling layers reduce dimensionality and increase generalization ability, and fully connected layers convert the extracted information into classification. Historically, models such as LeNet, AlexNet, VGG, and ResNet have been milestones in the development of CNNs. Today, CNNs are used in many fields such as image classification, object detection, face recognition, natural language processing, medicine, and agriculture. The advantages it provides include automatic feature extraction, high accuracy and parameter efficiency, while its disadvantages are high computational cost, the need for large datasets and

interpretation difficulties [18]. As a result, CNNs are considered to be one of the cornerstones of modern artificial intelligence, providing near-human level accuracy in visual data.

2.3. SqueezeNet Architecture

SqueezeNet is a compact convolutional neural network architecture developed by Forrest Iandola and his team in 2016. While large models like AlexNet offer high accuracy with the rise of deep learning, their millions of parameters have made their use difficult in mobile devices and embedded systems. Designed as a solution to this problem, SqueezeNet offers AlexNet-level accuracy while using approximately 50 times fewer parameters and reducing the model size to below 0.5 MB [19]. The most important innovation of SqueezeNet is its Fire Module structure. This module consists of two parts: the squeeze layer and the expand layer. The first layer reduces input parameters, while the second layer increases filtering capabilities. Thus, the model maintains its capacity to learn complex features while keeping its low parameter requirements. SqueezeNet's advantages include low memory consumption, fast operation, and reduced hardware requirements. Due to its small size, it is preferred in mobile devices, IoT systems, and applications requiring energy efficiency. It is also quite suitable for transfer learning [20]; it can be easily retrained for different tasks. However, larger and more complex architectures such as ResNet or EfficientNet can provide higher accuracy in some tasks [21]. SqueezeNet has demonstrated the applicability of deep learning in mobile and embedded systems with its lightweight but powerful structure. SqueezeNet was chosen for feature extraction from images because its architecture can generate deep features with high representational power at very low computational cost. SqueezeNet achieves a similar accuracy level to AlexNet while using approximately 50 times fewer parameters, significantly reducing memory and hardware requirements. Furthermore, its transfer learning-friendly structure allows for the efficient use of 1000 deep features extracted from potato disease images in machine learning algorithms such as ANN, KNN, SVM, and LR.

2.4. Artificial Neural Network (ANN)

Artificial Neural Networks (ANNs) are an artificial intelligence model developed by drawing inspiration from the functioning of neurons in the human brain. They basically consist of 3 layers: input, hidden, and output layers. The input layer receives the data, the hidden layers weight this data and process it through activation functions, and the output layer produces the final prediction. The learning process is based on error calculation and updating the weights with the backpropagation algorithm. The historical development of ANNs began with McCulloch & Pitts' artificial neuron

model in 1943, and learning networks emerged with Rosenblatt's perceptron in 1958 [22]. In the 1980s, the backpropagation algorithm made it possible to train multi-layer networks, and from the 2000s onwards, thanks to big data and powerful hardware, it paved the way for deep learning. Its advantages are flexibility, automatic learning, and high accuracy; The disadvantages are high computational costs, the need for large datasets, and interpretability challenges. Today, ANNs have become a powerful architecture at the heart of modern artificial intelligence, further enhanced by transfer learning and transformer-based models [23].

2.5. K-Nearest Neighbor (KNN)

K-Nearest Neighbor is one of the simplest and most understandable algorithms used in machine learning. It is among the supervised learning methods and its basic principle is based on the "similar data are close to each other" approach. It is a non-parametric and lazy learning algorithm; that is, it does not perform complex calculations during the training phase, all operations take place during prediction. The advantages of the algorithm include simplicity, flexibility, non-parametric structure, and high accuracy in small datasets. Its disadvantages are high computational cost in large datasets, memory dependence, scale sensitivity, and reduced performance in multidimensional data. KNN is used in many fields such as image processing, medicine, finance, and agriculture. Thanks to modern optimization techniques (KD-Tree, Ball-Tree, weighted KNN), it can also be used in large datasets. In conclusion, KNN is an indispensable algorithm in both academic and industrial applications with its simplicity and strong performance [24].

2.6. Support Vector Machine (SVM)

Support Vector Machine is used in supervised learning. Its main purpose is to find the hyperplane that best separates different classes in the dataset [25]. The points closest to this hyperplane are called "support vectors" and determine the decision boundary. SVM increases generalization ability by maximizing the margin between classes. Advantages of SVM include high accuracy, resilience to over-learning, and flexibility. Disadvantages include high computational cost in large datasets, critical kernel selection, and difficulty in hyperparameter tuning [26]. Its application areas are quite broad: it is successfully used in many fields such as spam filtering, sentiment analysis, bioinformatics, financial risk analysis, facial recognition, and medical diagnostics. In addition, "Support Vector Regression (SVR)" has been developed for regression problems. Among modern developments, multi-class SVM approaches (One-vs-One, One-vs-Rest), GPU-accelerated solutions, and integration with deep learning stand out [27]. In conclusion, SVM remains an important and powerful tool in both academic and

industrial applications.

2.7. Logistic Regression

Logistic Regression is one of the fundamental methods widely used in statistical modeling and machine learning to solve classification problems. This approach aims to model the relationship between independent variables and categorical dependent variables and to make probability estimations. Unlike linear regression, Logistic Regression produces limited probability values in the range of 0 to 1 instead of continuous values [28]. These probabilities are modeled through logit transformation, and the coefficients are determined using the Maximum Likelihood Estimation (MLE) method. The most common form of the model is binary classification (binary logistic regression), which separates two classes [29]. However, thanks to its multinomial and ordinal versions, it can also be applied to more complex classification problems. Logistic Regression has a wide range of applications, such as disease diagnosis in healthcare, credit risk analysis in the finance sector, predicting customer behavior in marketing, and spam email detection in natural language processing. The model's advantages include interpretability, computational efficiency, and the ability to make probability estimates [30]. However, its disadvantages include its inadequacy in nonlinear relationships and its lower performance in complex datasets.

2.8. Confusion Matrix

The confusion matrix is used to evaluate the prediction performance of training and test data. The values in the matrix are commonly used for performance measurement of classification problems. The confusion matrix of a two-class classification problem in the study is shown in Figure 2. The meanings of the rows and columns are as follows.

	Real Clas	
	Positive	Negative
Prediction Class	True TP	False TN
	True FP	False FN

Figure 2. Confusion Matrices

TP: True Positive. An example where the actual representation of the model is 1 and the predicted representation is also 1.

TN: True Negative. An example where the actual representation of the model is 0 and the predicted representation is also 0.

FP: False Positive. An example where the actual representation of the model is 0 and the predicted representation is 1.

FN: False Negative. Examples where the actual representation of the model is 1 and the predicted representation is 0.

2.9. Performance Metrics

Performance determination is used to evaluate the classifier performance. In this study, true positive (TP), true negative (TN), false positive (FP), and false negative (FN) values, as well as basic performance metrics such as accuracy, F-1 score, sensitivity, precision, and specificity, were calculated. The calculation formulas for these criteria are given below.

$$\frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

$$\frac{2TP}{2TP + FP + FN} \quad (2)$$

$$\frac{TP}{TP + FN} \quad (3)$$

$$\frac{TP}{TP + FP} \quad (4)$$

$$\frac{TN}{TN + FP} \quad (5)$$

3. EXPERIMENTAL RESULTS

In this study, ANN, KNN, SVM, and LR methods were used to classify the Potato Disease Dataset in order to detect potato diseases using image processing and machine learning methods, based on colony images obtained from potato producers during potato production. A computer with a Core i7™ 13620H 4.9 GHz processor, an NVIDIA GeForce RTX 4060 graphics card, and 64 GB of RAM was used. Python programming language was used. For the ANN method, the following parameters were used: 100

hidden layers, ReLu activation function, regularization 0.0001, number of iterations 200. For the KNN method, the following parameters were used: number of neighbors 5, metric euclidean, weight uniform. For the SVM method, the following parameters were used: cost 1, regression loss epsilon 0.1, kernel RBF, iteration 100. For the LR method, the regularization type Ridge L2 parameter was used. Training and testing were performed using the cross-validation method. In this method, the k value was set to 10. The procedures performed in the study are shown in Figure 3.

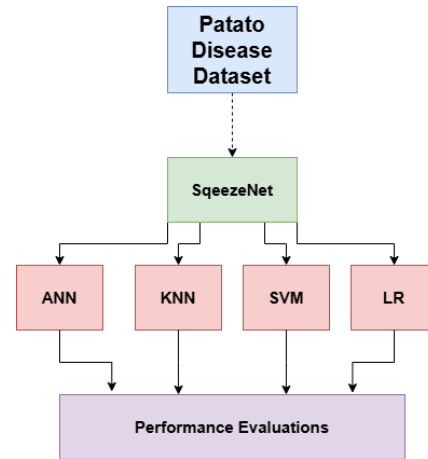


Figure 3. Flowchart

From each image obtained using the SqueezeNet architecture, 1000 features were given as input data to machine learning methods, and their training was carried out. The confusion matrices obtained from the machine learning methods are given in order. The confusion matrix obtained from the ANN method is shown in Figure 4.

Actual	Predicted								
		Black Scurf	Blackleg	Common Scab	Dry Rot	Healty Potatoes	Miscellaneous	Pink Rot	Σ
	Black Scurf	44	3	6	0	1	4	0	58
	Blackleg	2	29	3	14	0	6	6	60
	Common Scab	7	0	42	4	0	7	2	62
	Dry Rot	1	12	3	36	1	4	3	60
	Healty Potates	1	0	0	1	77	1	0	80
	Miscellaneous	5	11	6	7	2	41	2	74
	Pink Rot	1	7	1	3	1	4	40	57
	Σ	58	60	62	60	80	74	57	451

Figure 4. ANN Data

According to Figure 4, the ANN method correctly classified 44 data points belonging to the Black Scurf class. It classified 3 data points belonging to the Black Scurf class as Blackleg. It classified 6 data points belonging to the Black Scurf class as Common Scab. It classified 1 data point belonging to the Black Scurf class as Healthy Potatoes. It classified 4 data points belonging to the Black Scurf class as Miscellaneous. In total, it incorrectly classified 14 data points belonging to the Black Scurf class. It correctly classified 29 data points belonging

to the Blackleg class. It classified 2 data points belonging to the Blackleg class as Black Scurf. It classified 3 data points belonging to the Blackleg class as Common Scab. It classified 14 data points belonging to the Blackleg class as Dry Rot. It classified 6 data points belonging to the Blackleg class as Miscellaneous. It classified 6 data points belonging to the Blackleg class as Pink Rot. In total, it incorrectly classified 31 data points belonging to the Blackleg class. It correctly classified 42 data points belonging to the Common Scab class. It classified 7 data

points belonging to the Common Scab class as Black Scurf. It classified 4 data points belonging to the Common Scab class as Dry Rot. It classified 7 data points belonging to the Common Scab class as Miscellaneous. It classified 2 data points belonging to the Common Scab class as Pink Rot. In total, it incorrectly classified 20 data points belonging to the Common Scab class. It correctly classified 36 data points belonging to the Dry Rot class. It classified 1 data point belonging to the Dry Rot class as Black Scurf. It classified 12 data points belonging to the Dry Rot class as Blackleg. It classified 3 data points belonging to the Dry Rot class as Common Scab. It classified 1 data point belonging to the Dry Rot class as Healthy Potatoes. It classified 4 data points belonging to the Dry Rot class as Miscellaneous. It classified 3 data points belonging to the Dry Rot class as Pink Rot. In total, it incorrectly classified 24 data points belonging to the Dry Rot class. It correctly classified 77 data points belonging to the Healthy Potatoes class. It classified 1 data point belonging to the Healthy Potatoes class as Black Scurf. It classified 1 data point belonging to the Healthy Potatoes class as Dry Rot. It classified 1 data point belonging to the Healthy Potatoes class as Miscellaneous. In total, it incorrectly classified 3 data points belonging to the

Healthy Potatoes class. It correctly classified 41 data points belonging to the Miscellaneous class. It classified 5 data points belonging to the Miscellaneous class as Black Scurf. It classified 11 data points belonging to the Miscellaneous class as Blackleg. It classified 6 data points belonging to the Miscellaneous class as Common Scab. The study incorrectly classified 7 data points belonging to the Miscellaneous class as Dry Rot. It classified 2 data points belonging to the Miscellaneous class as Healthy Potatoes. It classified 2 data points belonging to the Miscellaneous class as Pink Rot. In total, it incorrectly classified 33 data points belonging to the Miscellaneous class. It correctly classified 40 data points belonging to the Pink Rot class. It classified 1 data point belonging to the Pink Rot class as Black Scurf. It classified 7 data points belonging to the Pink Rot class as Blackleg. It classified 1 data point belonging to the Pink Rot class as Common Scab. It classified 3 data points belonging to the Pink Rot class as Dry Rot. It classified 1 data point belonging to the Pink Rot class as Healthy Potatoes. It classified 4 data points belonging to the Pink Rot class as Miscellaneous. In total, it incorrectly classified 17 data points belonging to the Pink Rot class. Figure 5 shows the confusion matrix for the KNN method.

Actual	Predicted								Σ
		Black Scurf	Blackleg	Common Scab	Dry Rot	Healthy Potatoes	Miscellaneous	Pink Rot	
	Black Scurf	38	3	14	0	1	2	0	
	Blackleg	5	28	4	10	0	5	8	
	Common Scab	22	0	28	4	3	4	1	
	Dry Rot	4	14	12	23	2	4	1	
	Healthy Potatoes	1	0	0	0	78	0	1	
	Miscellaneous	6	9	8	5	11	34	1	
	Pink Rot	4	10	5	7	4	9	18	
	Σ	58	60	62	60	80	74	57	451

Figure 5 KNN Data

According to Figure 5, the KNN method correctly classified 38 data points belonging to the Black Scurf class. It classified 3 data points belonging to the Black Scurf class as Blackleg. It classified 14 data points belonging to the Black Scurf class as Common Scab. It classified 1 data point belonging to the Black Scurf class as Healthy Potatoes. It classified 2 data points belonging to the Black Scurf class as Miscellaneous. In total, it incorrectly classified 20 data points belonging to the Black Scurf class. It correctly classified 28 data points belonging to the Blackleg class. It classified 5 data points belonging to the Blackleg class as Black Scurf. It classified 4 data points belonging to the Blackleg class as Common Scab. It classified 10 data points belonging to the Blackleg class as Dry Rot. It classified 5 data points belonging to the Blackleg class as Miscellaneous. It classified 8 data points belonging to the Blackleg class as Pink Rot. In total, it incorrectly classified 32 data points belonging to the Blackleg class. It correctly classified 28 data points belonging to the Common Scab class. It classified 22 data

points belonging to the Common Scab class as Black Scurf. It classified 4 data points belonging to the Common Scab class as Dry Rot. It classified 3 data points belonging to the Common Scab class as Healthy Potatoes. It classified 4 data points belonging to the Common Scab class as Miscellaneous. It classified 1 data point belonging to the Common Scab class as Pink Rot. In total, it incorrectly classified 34 data points belonging to the Common Scab class. It correctly classified 23 data points belonging to the Dry Rot class. It classified 4 data points belonging to the Dry Rot class as Black Scurf. It classified 14 data points belonging to the Dry Rot class as Blackleg. It classified 12 data points belonging to the Dry Rot class as Common Scab. It incorrectly classified 37 data points in the Dry Rot category. It correctly classified 78 data points in the Healthy Potatoes category. It incorrectly classified 1 data point in the Healthy Potatoes category as Black Scurf. It incorrectly classified 1 data point in the Healthy Potatoes category as Pink Rot. In total, it incorrectly classified 2 data points in the Healthy Potatoes

category. It correctly classified 34 data points in the Miscellaneous category. It correctly classified 34 data points belonging to the Miscellaneous class. It classified 6 data points belonging to the Miscellaneous class as Black Scurf. It incorrectly classified 9 data points in the Miscellaneous category as Blackleg. It incorrectly classified 8 data points in the Miscellaneous category as Common Scab. The study incorrectly classified 40 data points belonging to the Miscellaneous class as Dry Rot. It correctly classified 18 data points belonging to the Pink Rot class. It classified 4 data points belonging to the Pink

Rot class as Black Scurf. It classified 10 data points belonging to the Pink Rot class as Blackleg. It classified 5 data points belonging to the Pink Rot class as Common Scab. It classified 7 data points belonging to the Pink Rot class as Dry Rot. It classified 4 data points belonging to the Pink Rot class as Healthy Potatoes. It classified 9 data points belonging to the Pink Rot class as Miscellaneous. In total, it incorrectly classified 39 data points belonging to the Pink Rot class. Figure 6 shows the confusion matrix for the SVM method.

Actual	Predicted								
		Black Scurf	Blackleg	Common Scab	Dry Rot	Healty Potatoes	Miscellaneous	Pink Rot	Σ
	Black Scurf	44	2	8	0	1	4	0	58
	Blackleg	2	35	3	8	0	5	7	60
	Common Scab	14	0	37	4	0	6	1	62
	Dry Rot	0	19	6	28	1	4	2	60
	Healty Potates	1	0	0	1	74	2	2	80
	Miscellaneous	5	12	6	4	3	40	4	74
	Pink Rot	1	6	0	3	1	8	38	57
	Σ	58	60	62	60	80	74	57	451

Figure 6. SVM Data

According to Figure 6, the SVM method correctly classified 44 data points belonging to the Black Scurf class. It classified 2 data points belonging to the Black Scurf class as Blackleg. It classified 6 data points belonging to the Black Scurf class as Common Scab. It classified 1 data point belonging to the Black Scurf class as Healthy Potatoes. It classified 4 data points belonging to the Black Scurf class as Miscellaneous. In total, it incorrectly classified 13 data points belonging to the Black Scurf class. It correctly classified 35 data points belonging to the Blackleg class. It classified 2 data points belonging to the Blackleg class as Black Scurf. It classified 3 data points belonging to the Blackleg class as Common Scab. It classified 8 data points belonging to the Blackleg class as Dry Rot. It classified 5 data points belonging to the Blackleg class as Miscellaneous. It classified 7 data points belonging to the Blackleg class as Pink Rot. In total, it incorrectly classified 25 data points belonging to the Blackleg class. It correctly classified 37 data points belonging to the Common Scab class. It classified 14 data points belonging to the Common Scab class as Black Scurf. It classified 4 data points belonging to the Common Scab class as Dry Rot. It classified 6 data points belonging to the Common Scab class as Miscellaneous. It classified 1 data point belonging to the Common Scab class as Pink Rot. In total, it incorrectly classified 25 data points belonging to the Common Scab class. It correctly classified 28 data points belonging to the Dry Rot class. It classified 19 data points belonging to the Dry Rot class as Blackleg. It classified 6 data points belonging to the Dry Rot class as Common Scab. It classified 1 data point belonging to the Dry Rot class as Healthy Potatoes. It

classified 4 data points belonging to the Dry Rot class as Miscellaneous. It misclassified 2 data points belonging to the Dry Rot class as Pink Rot. In total, it misclassified 32 data points belonging to the Dry Rot class. It correctly classified 74 data points belonging to the Healthy Potatoes class. It classified 1 data point belonging to the Healthy Potatoes class as Black Scurf. It classified 1 data point belonging to the Healthy Potatoes class as Dry Rot. It classified 2 data points belonging to the Healthy Potatoes class as Miscellaneous. It classified 2 data points belonging to the Healthy Potatoes class as Pink Rot. In total, it misclassified 6 data points belonging to the Healthy Potatoes class. It correctly classified 40 data points belonging to the Miscellaneous class. It classified 5 data points belonging to the Miscellaneous class as Black Scurf. It classified 12 data points belonging to the Miscellaneous class as Blackleg. It classified 6 data points belonging to the Miscellaneous class as Common Scab. The LR method incorrectly classified 34 data points belonging to the Miscellaneous class as Dry Rot. It also correctly classified 38 data points belonging to the Pink Rot class. In total, it incorrectly classified 34 data points belonging to the Miscellaneous class. It correctly classified 38 data points belonging to the Pink Rot class. It classified 1 data point belonging to the Pink Rot class as Black Scurf. It classified 6 data points belonging to the Pink Rot class as Blackleg. It classified 3 data points belonging to the Pink Rot class as Dry Rot. It classified 1 data point belonging to the Pink Rot class as Healthy Potatoes. In total, it incorrectly classified 19 data points belonging to the Pink Rot class. Figure 7 shows the confusion matrix for the LR method.

Actual	Predicted							
		Black Scurf	Blackleg	Common Scab	Dry Rot	Healty Potatoes	Miscellaneous	Pink Rot
	Black Scurf	40	5	9	1	0	3	0
	Blackleg	4	28	3	9	0	10	6
	Common Scab	10	0	42	3	0	6	1
	Dry Rot	0	9	4	35	1	10	1
	Healty Potates	1	0	2	1	74	2	0
	Miscellaneous	4	9	5	8	4	40	4
	Pink Rot	0	5	1	3	0	5	43
Σ		58	60	62	60	80	74	57
								451

Figure 7. LR Data

According to Figure 7, the LR method correctly classified 40 data points belonging to the Black Scurf class. It classified 5 data points belonging to the Black Scurf class as Blackleg. It classified 9 data points belonging to the Black Scurf class as Common Scab. It classified 1 data point belonging to the Black Scurf class as Dry Rot. It classified 3 data points belonging to the Black Scurf class as Miscellaneous. In total, it incorrectly classified 18 data points belonging to the Black Scurf class. It correctly classified 28 data points belonging to the Blackleg class. It classified 4 data points belonging to the Blackleg class as Black Scurf. It classified 3 data points belonging to the Blackleg class as Common Scab. It classified 9 data points belonging to the Blackleg class as Dry Rot. It classified 10 data points belonging to the Blackleg class as Miscellaneous. It classified 6 data points belonging to the Blackleg class as Pink Rot. In total, it incorrectly classified 25 data points belonging to the Blackleg class. It correctly classified 42 data points belonging to the Common Scab class. It classified 10 data points belonging to the Common Scab class as Black Scurf. It classified 3 data points belonging to the Common Scab class as Dry Rot. It classified 6 data points belonging to the Common Scab class as Miscellaneous. It classified 1 data point belonging to the Common Scab class as Pink Rot. In total, it incorrectly classified 20 data points belonging to the Common Scab class. It correctly classified 35 data points belonging to the Dry Rot class. It classified 9 data points belonging to the Dry Rot class as Blackleg. It classified 4 data points belonging to the Dry Rot class as Common Scab. It classified 1 data point belonging to the Dry Rot class as Healthy Potatoes. It classified 10 data points belonging to the Dry Rot class as Miscellaneous. It incorrectly classified 1 data point belonging to the Dry Rot class as Pink Rot. In total, it incorrectly classified 25 data points belonging to the Dry Rot class. It correctly classified 74 data points belonging to the Healthy Potatoes class. It classified 1 data point belonging to the Healthy Potatoes class as Black Scurf. It classified 2 data points belonging to the Healthy Potatoes class as Common Scab. It classified 1 data point belonging to the Healthy Potatoes class as Dry Rot. It classified 2 data points belonging to the Healthy Potatoes class as Miscellaneous. In total, it incorrectly classified 6 data

points belonging to the Healthy Potatoes class. It correctly classified 40 data points belonging to the Miscellaneous class. It classified 4 data points belonging to the Miscellaneous class as Black Scurf. It classified 9 data points belonging to the Miscellaneous class as Blackleg. It classified 5 data points belonging to the Miscellaneous class as Common Scab. The data sample incorrectly classified 8 data points belonging to the Miscellaneous class as Dry Rot. It classified 4 data points belonging to the Miscellaneous class as Healthy Potatoes. It classified 4 data points belonging to the Miscellaneous class as Pink Rot. In total, it incorrectly classified 25 data points belonging to the Miscellaneous class. It correctly classified 43 data points belonging to the Pink Rot class. It classified 5 data points belonging to the Pink Rot class as Blackleg. It classified 1 data point belonging to the Pink Rot class as Common Scab. It classified 3 data points belonging to the Pink Rot class as Dry Rot. It classified 5 data points belonging to the Pink Rot class as Miscellaneous. In total, it incorrectly classified 14 data points belonging to the Pink Rot class. The graph of the performance metrics is given in Figure 8.

Model	Accuracy	F1 Score	Precision	Recall
ANN	0.685	0.685	0.686	0.685
KNN	0.548	0.536	0.548	0.548
SVM	0.656	0.660	0.660	0.656
LR	0.670	0.670	0.670	0.670

Figure 8. Performance Metrics Data

Examining Figure 8, the ANN model has the highest classification success, while the KNN model has the lowest. The classification success of the LR model is 0.670, and the classification success of the SVM model is 0.656. When other performance metrics are examined, it is seen that the F1 Score, Precision, and Recall metric values show parallelism with the classification success of the models. When the classification successes and F1 Score, Precision, and Recall metrics are examined, it can be said that all models show a success rate of 54.8% and above. One of the main reasons for the classification success rates falling below expectations is that the dataset consists of

only 451 images, resulting in a relatively limited sample size for deep learning-based feature extraction. Furthermore, some disease classes in the dataset exhibited visually very similar symptoms, leading to confusion between classes. The fact that the images were obtained at different resolutions and under different conditions also increased within-class variation, negatively impacting model performance. In addition, the features obtained from SqueezeNet in this study were classified using classical machine learning algorithms, and the class discrimination power that could be provided by deeper, end-to-end trained networks was not fully achieved. The study's seven-class problem presented a more challenging classification scenario compared to commonly used binary classification problems in the literature, thus lowering success rates.

4. Conclusion

Image processing methods have become frequently used in identifying potato diseases recently. Accurate analysis of the data obtained from these methods allows for the rapid determination of potato production quality. Based on this, this study classified data obtained from 451 different potato photographs. The highest classification success rate was obtained from the ANN model at 68.5%. The lowest classification success rate was obtained from the KNN model at 54.8%.

Examining the results, it can be said that the machine learning models used in this study can be used in potato disease classification problems. Data from 451 different potato samples were used. In this case, the success of the models may decrease in some real-life tests. To overcome this problem, the number of data points can be increased. In addition to increasing the number of data points, higher results can be obtained in real-life tests using different classification methods.

With the classification models proposed in this study and the data obtained from sensors, potato quality can be quickly determined in production facilities, harvesting networks, packaging facilities, and businesses where potatoes are consumed. This will enable consumers to consume potatoes of the highest quality.

Declaration of Ethical Standards

This research does not involve direct interaction with human participants or animals. Accordingly, ethical approval and informed consent were not required.

Credit Authorship Contribution Statement

All authors contributed substantially and equally to this study. The processes of conceptualization, methodology development, software implementation, validation, formal analysis, investigation, data curation, manuscript drafting, review and editing, as well as visualization were carried out collaboratively. All authors have reviewed and

approved the final version of the manuscript.

Declaration of Competing Interest

The authors declare that there are no conflicts of interest regarding the publication of this manuscript.

Funding / Acknowledgements

This research received no external funding.

Data Availability

The dataset used in this study was obtained from a publicly available source and subsequently processed by the authors for classification purposes. Further details regarding the processed dataset are available from the corresponding author upon reasonable request.

References

- [1] Arshaghi, A., M. Ashourian, and L. Ghabeli, *Potato diseases detection and classification using deep learning methods*. Multimedia Tools and Applications, 2023. **82**(4): p. 5725–5742.
- [2] Singh, A. and H. Kaur. *Potato plant leaves disease detection and classification using machine learning methodologies*. in *IOP Conference Series: Materials Science and Engineering*. 2021. IOP Publishing.
- [3] Oishi, Y., et al., *Automated abnormal potato plant detection system using deep learning models and portable video cameras*. International Journal of Applied Earth Observation and Geoinformation, 2021. **104**: p. 102509.
- [4] Alzakari, S.A., et al., *Early detection of potato disease using an enhanced convolutional neural network-long short-term memory deep learning model*. Potato Research, 2025. **68**(1): p. 695–713.
- [5] Zhang, F., et al. *Hyperspectral imaging combined with convolutional neural network for outdoor detection of potato diseases*. in *2021 6th International Symposium on Computer and Information Processing Technology (ISCIPIT)*. 2021. IEEE.
- [6] Kumar, A. and V.K. Patel, *Classification and identification of disease in potato leaf using hierarchical based deep learning convolutional neural network*. Multimedia Tools and Applications, 2023. **82**(20): p. 31101–31127.
- [7] Anim-Ayeko, A.O., C. Schillaci, and A. Lipani, *Automatic blight disease detection in potato (*Solanum tuberosum* L.) and tomato (*Solanum lycopersicum*, L. 1753) plants using deep learning*. Smart Agricultural Technology, 2023. **4**: p. 100178.
- [8] Suarez Baron, M.J., A.L. Gomez, and J.E.E. Diaz, *Supervised learning-based image classification for the detection of late blight in potato crops*. Applied Sciences, 2022. **12**(18): p. 9371.
- [9] Tarik, M.I., et al. *Potato disease detection using machine learning*. in *2021 Third international conference on intelligent communication technologies and virtual mobile networks (ICICV)*. 2021. IEEE.
- [10] Bienkowski, D., et al., *Detection and differentiation between potato (*Solanum tuberosum*) diseases using calibration models trained with non-imaging spectrometry data*. Computers and Electronics in Agriculture, 2019. **167**: p. 105056.
- [11] Kadam, S.U., et al., *Machine learning methode for automatic potato disease detection*. NeuroQuantology, 2022. **20**(16): p. 2102–2106.
- [12] Iqbal, M.A. and K.H. Talukder. *Detection of potato disease using image segmentation and machine learning*. in *2020 international conference on wireless communications signal processing and networking (WiSPNET)*. 2020. IEEE.
- [13] Islam, M., et al. *Detection of potato diseases using image segmentation and multiclass support vector machine*. in *2017 IEEE 30th canadian conference on electrical and computer engineering (CCECE)*. 2017. IEEE.
- [14] Pinki, F.T., N. Khatun, and S. Islam, *Visual features based paddy leaf disease recognition, its severity detection and remedy prediction using k-means clustering and adaboost*.

- Journal of Image Processing & Pattern Recognition Progress, 2020. 7(3): p. 41–52.
- [15] Faria, F.T.J., et al. *Classification of potato disease with digital image processing technique: a hybrid deep learning framework*. in *2023 IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC)*. 2023. IEEE.
- [16] Potato Diseases Datasets. <https://doi.org/10.34740/kaggle/dsv/7203264> on 27 October 2025
- [17] Zeiler, M.D. and R. Fergus. *Visualizing and understanding convolutional networks*. in *European conference on computer vision*. 2014. Springer.
- [18] Samek, W., T. Wiegand, and K.-R. Müller, *Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models*. arXiv preprint arXiv:1708.08296, 2017.
- [19] Iandola, F.N., et al., *SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and < 0.5 MB model size*. arXiv preprint arXiv:1602.07360, 2016.
- [20] Pan, S.J. and Q. Yang, *A survey on transfer learning*. IEEE Transactions on knowledge and data engineering, 2009. 22(10): p. 1345–1359.
- [21] Tan, M. and Q. Le. *Efficientnet: Rethinking model scaling for convolutional neural networks*. in *International conference on machine learning*. 2019. PMLR.
- [22] Strawn, G., *Where Deep Learning and Generative AI Started: Masterminds of Artificial Neural Networks—McCulloch, Pitts, and Rosenblatt*. IT Professional, 2024. 26(03): p. 99–101.
- [23] Tüfekçi, M. and F. Karpat. *Derin Öğrenme Mimarilerinden Konvolüsyonel Sinir Ağları (CNN) Üzerinde Görüntü İşleme-Sınıflandırma Kabiliyetinin Arttırılmasına Yönelik Yapılan Çalışmaların İncelenmesi*. in *International Conference on Human-Computer Interaction, Optimization and Robotic Applications*. 2019.
- [24] Wang, L. *Research and implementation of machine learning classifier based on KNN*. in *IOP Conference Series: Materials Science and Engineering*. 2019. IOP publishing.
- [25] Vapnik, V., *Support-vector networks*. Machine learning, 1995. 20: p. 273–297.
- [26] Hsu, C.-W. and C.-J. Lin, *A comparison of methods for multiclass support vector machines*. IEEE transactions on Neural Networks, 2002. 13(2): p. 415–425.
- [27] Catanzaro, B., N. Sundaram, and K. Keutzer. *Fast support vector machine training and classification on graphics processors*. in *Proceedings of the 25th international conference on Machine learning*. 2008.
- [28] Agresti, A., *Analysis of ordinal categorical data*. 2010: John Wiley & Sons.
- [29] Hosmer Jr, D.W., S. Lemeshow, and R.X. Sturdivant, *Applied logistic regression*. 2013: John Wiley & Sons.
- [30] Menard, S., *Applied logistic regression analysis*. 2001: SAGE publications.

*Research Article***Performance Analysis of Machine Learning Algorithms on Wisconsin Diagnostic Breast Cancer Data Set Enriched with Data Augmentation Technique****Ayça ACET^a , Abdullah Erhan AKKAYA^{b,*}** ^a Turkish Aerospace Industry, 06980, Kahramankazan, Ankara, Türkiye^b İnönü University, Faculty of Engineering, Department of Computer Engineering, Malatya, Türkiye

ARTICLE INFO

Article history:

Received 14 February 2026

Accepted 26 June 2026

*Keywords:*Breast cancer diagnosis,
Classification,
Data augmentation,
Machine learning

ABSTRACT

Breast cancer is a life-threatening cancer worldwide and is commonly seen in women. Early and accurate diagnosis is a key to successful treatment and better survival rates. In this study, using 569 samples from the Wisconsin Diagnostic Breast Cancer dataset, the classification performances of Support Vector Machines (SVM), Adaptive Boosting (AdaBoost), Random Forest (RF), Naïve Bayes (NB) and k-Nearest Neighbor (kNN) algorithms are compared. To improve accuracy, the train dataset was expanded from 381 to 1524 samples using a three-stage method involving random scaling, Gaussian noise, and mathematical transformations. Test dataset remained unchanged. Statistical analyses confirmed that the augmented data presented sufficient variability to enhance generalization while preserving the original distribution. Model performance analysis was conducted using accuracy, precision, sensitivity, F1 score, FDR, MCC and AUC values. Compared to the original dataset, the accuracy of the SVM algorithm improved from 98.94% to 99.47% with the data augmentation techniques. This result shows that data augmentation techniques are effective in improving the classification performance of machine learning models for breast cancer diagnosis.

This is an open access article under the CC BY-SA 4.0 license.
(<https://creativecommons.org/licenses/by-sa/4.0/>)

1. Introduction

The regulation of cell proliferation in healthy tissues is guided by responsive checkpoint networks rather than a static, predetermined schedule. Instead of following a rigid, automated script, cellular growth decisions require the continuous integration of a wide array of inputs, effectively bridging intrinsic transcriptional states with extrinsic microenvironmental cues. Mechanistically, this integration relies on the joint actions of layered transcription factor cascades [1-3], epigenetic machinery [4, 5], localized cytokine signaling [6, 7], and homeostatic metabolic sensors [8, 9]. Because these pathways converge to act as a collective processor, the resulting system is inherently flexible-allowing cellular behavior to shift dynamically based on the cumulative environmental and

internal context. Understanding how these diverse signals are synthesized to maintain this delicate balance between proliferation and quiescence remains a foundational challenge in tissue biology.

However, mutations and abnormal gene changes can cause cells to divide and multiply uncontrollably. Over-proliferation causes cells to come together to form solid tissue called a tumor. If a tumor grows only at the place it develops, then it is a benign tumor. On the other hand, if a tumor spreads to different tissues, then it is a malignant tumor [10, 11]. Malignant tumors are also called cancerous tumors; the cells they contain are called cancer cells, Figure 1. The benign stage of a tumor is usually not life-threatening, while the malignant stage can be fatal.

A cancerous tumor located in the breast-an essential

* Corresponding author. E-mail address: erhan.akkaya@inonu.edu.tr
DOI: 10.58190/ijamec.2026.171

organ of the female reproductive system-is called breast cancer. This devastating condition is highly influenced by numerous risk factors that can be categorized into two major groups: reproductive and non-reproductive risk factors. The risk factors in the reproductive aspects that predispose a woman to breast cancer are many. First, menarche at an early age has been associated with an increased risk of the development of malignant tumors [12]. Also, late menopause, the natural cessation of menstruation in older age, has been associated with a higher susceptibility to breast cancer [12]. Additionally, factors such as having children at an older age [13] or never having had children [14], as well as breastfeeding less frequently [15], have been identified as potential reproductive risk factors.

Non-reproductive risk factors, on the other hand, are relevant to breast cancer etiology [16]. Of all these risk factors, the main ones are genetic BRCA1 and BRCA2 mutations [17]. People who carry these gene mutations have a much higher risk of developing the disease than the general population. [18]. Obesity [19], considered to be a state characterized by an excess of body weight and/or adipose tissue, has also been implicated as a risk factor contributing etiopathogenetically to the genesis of breast cancer. Postmenopausal obesity is a third non-reproductive risk factor that increases susceptibility to this malignancy. In addition, having a family history of breast cancer is also known as a non-reproductive risk factor since women with blood relatives (mother, sister(s), and children) are more likely to get the disease [19]. Among other significant risk factors are hormone disorders [20], radiation therapy, and alcohol. Only in 2020, the global number of newly diagnosed cancer cases directly due to alcohol consumption was 741,300 [21]. This frightful data makes up 4% of all cancers in the world. People's lifestyle decisions directly affect their individual health. The most striking example of this is the direct link between alcohol and breast cancer.

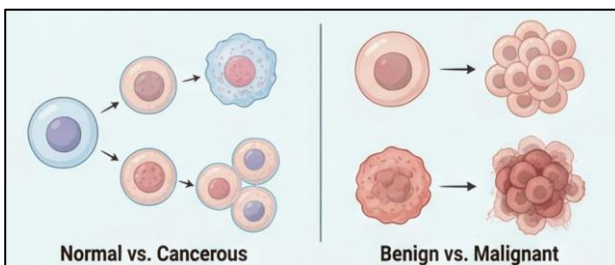


Figure 1. Cellular proliferation, healthy and tumor cells.

Figure 1 shows healthy and cancerous cells. The upper left shows a normal cell's healthy division and differentiation cycle. The lower left shows cancerous cells that have multiplied uncontrollably due to a disruption in the differentiation cycle. The upper right shows a benign tumor structure where the cells are clustered within a defined boundary. The lower right shows a malignant

(cancerous) tumor structure where the cells have undergone morphological deformation and spread to surrounding tissues.

The World Health Organization (WHO) has estimated that in the year 2022, approximately 2.3 million new patients were diagnosed with breast cancer worldwide. Unfortunately, this number accounts for 11.6% of all cancer cases diagnosed in 2022. Breast cancer was the most common cancer in women in 157 out of 185 countries in 2022 [22]. In less developed countries, more than half of breast cancer cases occur in women under the age of 50, while in developed countries like the UK, more than a third of cases occur in women over the age of 70 [23]. According to Globocan, 25,249 new cases of breast cancer were diagnosed in Turkey in 2022. This accounts for 23.5% of all female cancers [24]. Early diagnosis and treatment of breast cancer save lives. The Turkish Breast Cancer Foundation, the American College of Radiology, and the National Comprehensive Cancer Network-NCCN recommend that women should start mammography screening at age 40 and continue their screening every two years up to the age of 69. This recommendation is consistent with several international guidelines advocating early detection of breast cancer with regular mammography [25, 26]. Studies show that regular mammography screening reduces deaths from breast cancer [27]. However, it has been reported that participation in mammography screening among women living in Türkiye is low [26, 28].

Diagnosing benign and malignant tumor cells is crucial in determining the type of treatment. To achieve this diagnosis, this study focuses on five commonly used classification algorithms: Adaptive Boosting (AdaBoost), k-Nearest Neighbor (kNN), Naïve Bayes (NB), Random Forest (RF), and Support Vector Machines (SVM). These algorithms were used to classify cancer cells based on the provided cellular data, thereby predicting the type of cancer. To evaluate the performance and reliability of the classification algorithms, the Wisconsin Breast Cancer Dataset from the University of California, Irvine (UCI) data mining repository was used. The WDBC dataset contains thirty characteristics of breast cancer cells from 569 patients. The WDBC dataset was chosen for this study because it is lossless.

This study is built upon and significantly extends the preliminary machine learning models developed in the author's Master's thesis [29]. While that previous work focused solely on evaluating the baseline performance of these models on the original dataset, this paper introduces a novel three-stage data augmentation strategy to enhance classification robustness and generalize across small medical datasets. To present these contributions clearly, the paper is structured into six parts: The first part provides a general introduction to the problem. The second part is devoted to solution methods previously applied to the

WDBC dataset in the literature. The third part begins by describing the WDBC dataset in detail, and continues with data augmentation techniques and hyperparameter optimization of machine learning methods. The fourth part presents a comparative analysis of the performance of AdaBoost, kNN, NB, RF, and SVM algorithms in the context of breast cancer diagnosis. The fifth part provides information about the limitations of the study. Lastly, the sixth part concludes the paper and outlines future work.

2. RELATED WORKS

WDBC dataset used in this article consists of 569 samples and 30 features. In this section, the results of the studies that classify the WDBC dataset using AdaBoost, kNN, NB, RF, and SVM methods are given. In the study conducted by Cabanillas-Carbonell and Zapata-Paulini [30], it was observed that AdaBoost 0.97, NB 0.91 and RF had 0.97 accuracies. In another study [31], SVM and RF algorithms gave correct classification rates of 0.986 and 0.965, respectively. Alzboon et al. [32] reported accuracy rates of 0.914 for AdaBoost, 0.807 for kNN, 0.935 for NB, 0.949 for RF and 0.975 for SVM. In a study conducted by Jakhar et al. [33], it was shown that AdaBoost and RF had accuracies of 0.9724 and 0.97. In addition, Chen et al. [34] obtained accuracies of 0.912 and 0.965 for kNN and RF models, respectively. Özcan et al. applied eleven machine learning models [35]. Among the AdaBoost, kNN, NB, RF, and SVM methods, the accuracy rate has been reported to be 0.9806, 0.9385, 0.9737, 0.9736, and 0.9766, respectively. During a similar study, Mashudi et al. [36] assessed five models, with the recorded values being 0.9877.

Imran et al. [37] observed an accuracy of 0.98 when using Random Forest. Random Forest had been previously shown to outperform Naive Bayes and AdaBoost classifiers. They further reported AdaBoost to be at 0.9824, leading the comparative classifiers. They also reported that Naive Bayes could achieve an accuracy of approximately 0.9619. Prastyo et al. [38] compared eight different machine learning models with the application of AdaBoost, kNN, RF, and SVM methods. These resulted in accuracy rates of 0.9561, 0.9649, 0.9561, and 0.9561, respectively. In the work by Aamir et al. [39], five different machine-learning algorithms were used for the classification process. For the 60:40 ratio, the accuracy values in RF and SVM were 0.954 and 0.9755, respectively. When the training and test datasets ratio is 70:30, the accuracy values for RF and SVM are 0.9667 and 0.9721, respectively. When the training and test datasets ratio is 80:20, the accuracy values for RF and SVM are 0.9807 and 0.9776, respectively.

Nuzhat et al. [40] tested six basic algorithms for breast cancer classification on the WDBC dataset. In this context, they applied the KNN, SVM, Logistic Regression, RF,

GBM, and XGBoost methods. The researchers developed a new ensemble architecture by integrating these classifiers using the Hard Voting technique. In the performance analysis of individual models, GBM demonstrated the highest success, achieving an accuracy of 98.33% and a recall of 100%. However, the ensemble model-composed of a combination of SVM, RF, and GBM components-achieved an accuracy of 98.89%, precision of 98.08%, recall of 100%, and an F1 score of 99.03%. This developed hybrid structure achieved a higher success rate than individual models across all evaluation metrics [40].

Pristyanto and Wirantanu [41] conducted a comparative analysis of Recursive Feature Elimination (RFE) and Genetic Algorithm (GA)-based feature selection methods on the WDBC dataset, which consists of 30 features and 569 data points. Logistic Regression, SVM, Naive Bayes, KNN, AdaBoost, and Random Forest (RF) algorithms were tested in the classification process. Accuracy analysis revealed that GA-based feature selection outperformed RFE in terms of weighted macro precision, recall, and F1 score metrics.

Tawil et al. [42] analyzed SVM, LightGBM, Random Forest (RF), Logistic Regression (LR), k-NN, and Naive Bayes algorithms on the WDBC dataset. These classifiers were subjected to a comprehensive comparative performance evaluation by integrating them with three different feature selection methods: Pearson correlation coefficient, Lasso, and mRMR. In the baseline scenario without a feature selection filter, LightGBM demonstrated the highest performance with an accuracy rate of 95%. In the mRMR-based feature selection architecture, where the number of features was reduced to 15, LightGBM's classification accuracy rose to 98%. On the other hand, in the Lasso feature selection model, where dimensionality was reduced to 5 features, the LR algorithm recorded the highest performance with an accuracy of 96% [42].

Mezaghrani et al. [43] proposed a hybrid feature selection method (CMGWO) that integrates correlation analysis with the Modified Gray Wolf Optimization (MGWO) algorithm to perform accurate breast cancer classification on the WDBC dataset. The performance of the developed approach was empirically tested using Random Forest (RF), SVM, and Naive Bayes classifiers. The experimental results show that the RF classifier achieved an accuracy rate of 99.12% with the CMGWO-based feature selection architecture, outperforming other alternative approaches in literature.

Salhi et al. [44] evaluated the performance of KNN, RF, MLP, LR, and SVM classifiers on three different datasets, including the WDBC dataset. In this process, they proposed innovative hybrid optimization methods-such as Two-Stage Mutated Gray Wolf Optimization (TMGWO), the Improved Salp Swarm Algorithm (ISSA), and Binary Black Particle Swarm Optimization (BBPSO)-to reduce dimensionality and improve model performance. In the

baseline scenario, where no feature selection procedure was applied, the KNN algorithm achieved a 90% accuracy rate using all 30 features in the WDBC dataset. On the other hand, the proposed TMGWO-based hybrid approach demonstrated the best performance in both feature optimization (feature selection) and classification accuracy compared to the other methods experimentally compared.

Zhu et al. [45] presented an innovative feature selection approach that integrates the Recursive Feature Elimination (RFE) protocol based on SHAP (Shapley Additive exPlanations) values for the detection of early-stage breast cancer on the WDBC dataset. During the experimental process, KNN, RF, LR, SVM, and LightGBM models were evaluated; the hyperparameter optimization of the classifiers was performed using the Particle Swarm Optimization (PSO) algorithm. Through the proposed hybrid algorithm, 26 critical features were filtered from the dataset. The optimized LightGBM-PSO model achieved 99.0% accuracy, 100% specificity, 100% precision, 97.40% recall, an F-measure of 98.68, and an AUC of 0.9870, demonstrating superior performance compared to the literature.

Ayepeku [46] applied Logistic Regression (LR), KNN, Support Vector Classifier (SVC), Decision Trees (DT), Random Forest (RF), Gradient Boosting (GBM), AdaBoost, and XGBoost algorithms on a comprehensive dataset containing clinical features obtained from the Kaggle platform. The diagnostic performance of the classification models was evaluated using the precision, recall, F1 score, and accuracy metrics. In the performance analysis, when the LR, KNN, SVC, and RF models were compared, it was observed that the accuracy values ranged between 0.91 and 0.93.

3. DATASET, PERFORMANCE METRICS and ANALYSIS TOOL

This section discusses the dataset used in this study and the evaluation metrics used for the analysis. The data description section includes the source and structure of the dataset and its statistical properties. Under the heading of performance metrics, this study discusses accuracy, sensitivity, precision, F1-score, false discovery rate, Matthew's correlation coefficient and area under the curve metrics. These metrics help to determine how well the models perform in classifying or predicting outcomes based on the given data. The hyperparameter optimization strategies and the baseline training configurations for the five machine learning algorithms (AdaBoost, kNN, NB, RF, and SVM) were adopted from [29].

3.1. Dataset Description

This study uses the Wisconsin Diagnostic Breast Cancer dataset from the University of California - Irvine Machine Learning Repository. The breast cancer dataset was

created by Dr. William H. Wolberg of the Department of General Surgery and W. Nick Street and Olvi L. Mangasarian of the Department of Computer Science at the University of Wisconsin in 1995. The dataset contains ten essential attributes: radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry, and fractal dimension. The dataset consists of 10 mean values, 10 standard error values, and ten maximum values of the basic attributes of breast cancer cells examined under the microscope. Together with the patient ID information and class label, the dataset consists of 569 samples with 30 attributes, plus an ID and a class label (total 32 columns). Among the 569 samples in the dataset are 357 benign and 212 malignant breast cancer cells. In WDBC dataset there are no attributes with missing information. Table 1 summarizes the minimum and maximum values of the type, mean, standard deviation, and maximum attributes of the WDBC dataset.

Table 1. Characteristics of the Wisconsin Diagnostic Breast Cancer (WDBC) dataset.

#	Feature	Feature Type	Mean (min-max)	Standard Error (min-max)	Worst (min-max)
1	Patient ID No.	Numeric	–	–	–
2	Diagnosis	Binary	–	–	–
3	Radius	Numeric	6.981 – 28.11	0.112 – 2.873	7.93 – 36.04
4	Texture	Numeric	9.71 – 39.28	0.36 – 4.89	12.02 – 49.54
5	Perimeter	Numeric	43.79 – 188.5	0.76 – 21.98	50.41 – 251.20
6	Area	Numeric	143.5 – 2501.0	6.80 – 542.20	185.2 – 254.0
7	Smoothness	Numeric	0.053 – 0.163	0.002 – 0.031	0.071 – 0.223
8	Compactness	Numeric	0.019 – 0.345	0.002 – 0.135	0.027 – 1.058
9	Concavity	Numeric	0.000 – 0.427	0.000 – 0.396	0.000 – 1.252
10	Concave points	Numeric	0.000 – 0.201	0.000 – 0.053	0.000 – 0.291
11	Symmetry	Numeric	0.106 – 0.304	0.008 – 0.079	0.157 – 0.664
12	Fractal dimension	Numeric	0.050 – 0.097	0.001 – 0.030	0.055 – 0.208

The radius value in the dataset is the radius information of the cells. The radius feature ranges from 6.981 to 28.110 on average. This shows that there are significant variations in cell sizes. This variation in cell size is an essential factor in diagnosing breast cancer. The texture is the rate of change in the grayscale of the inner surfaces of the cells. Texture characterization is vital in distinguishing breast cancer cells from normal cells. Perimeter is the value based on the radius of each cell. The perimeter has the widest range (43.79 - 188.5). The wide range indicates a great variation in the shape and size of the tumors. This variation is a factor to be considered in the diagnosis of breast cancer. Area is the surface area of the cell. The area

features also has a wide range (143.5 - 2501.0). This shows that there is diversity from very small cells to very large cells. Smoothness is the value of the radius of neighboring cells around the cell, i.e. the local variation in radius lengths. Compactness is the average cell density obtained by subtracting the square of the perimeter length of the cell divided by the cell area. Concavity is the magnitude of the indentations and protrusions around the cell. Concave points are the number of indentation and protrusion points around the cell. Smoothness, compactness, concavity, and concave points properties are generally close to zero and measure the surface irregularity of the cells. Symmetry is the value of the elliptical shape change of the cells. The fractal dimension is the ratio of nested irregular cells to all normal cells. The symmetry and fractal dimension properties describe the geometric properties of the cells and vary over a relatively narrow range. Finally, the standard deviation values are quite high, especially for area and perimeter length, indicating large variations in these properties.

3.2. Data Augmentation

In this study, data augmentation techniques were applied to the Wisconsin Diagnostic Breast Cancer (WDBC) dataset. Each observation in the original WDBC dataset contains 30 numerical features. To prevent data leakage, the original dataset was first partitioned into training and test subsets, and data augmentation was applied only to the training data. The partitioning ratio was 67% and 33% for the training and test datasets, respectively. Data augmentation was not applied to 188 test datasets out of 569 data. The 381 training datasets were expanded in three ways. Gaussian noise was added to the training dataset via $\pm 5\%$ random variations. Random scaling within a $\pm 10\%$ threshold was applied to the training dataset to capture natural variance. New samples were created in the training dataset through inter-feature mathematical operations. As a result, the original training set was expanded to 1,524 samples, including the original 381 samples. The test dataset remained unchanged. The augmented training samples were designed to preserve the original statistical properties of the data. To maintain physiological realism, the resulting negative feature values were reduced to zero. Statistical validation of the data augmentation process is presented in Figure 2. Histogram analysis based on probability density functions (PDFs) shows that the augmented training data preserves the basic distributional features of the original training set, including mode, skewness, and kurtosis. The observed overlap between the original and augmented distributions indicates that the introduced variations do not distort the basic feature space.

3.3. Analysis Tool

Gentleman and Ihaka originally programmed R. Since about mid-1997 there has been a development core team with write access to the R source [47]. The R language provides an extensive range of statistical and graphical techniques. R language offers methods such as clustering, classification, linear and nonlinear modeling and time series analysis. The R language is also developed as a complete programming language with features to enable users to add new functionality by writing R functions [48]. R users in Turkey can download the R software from <http://cran.pau.edu.tr/>, the local CRAN server hosted at Pamukkale University. R is used by academic research organizations and many large companies such as Uber, Google, Airbnb, and Facebook [49]. Despite the widespread use of R in industry, academics are the largest user group of the R language.

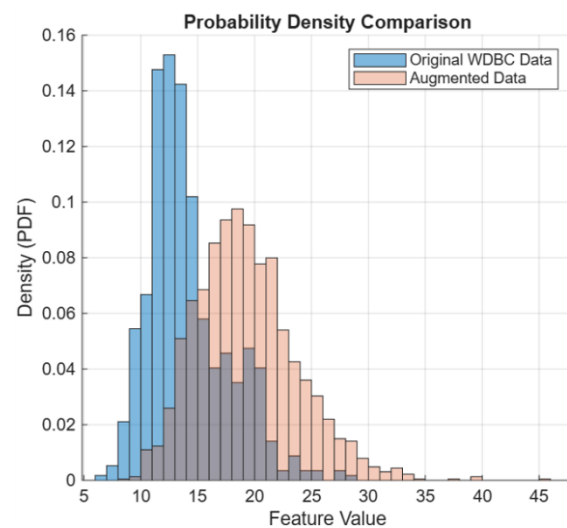


Figure 2. Comparative probability density analysis of original ($n=569$) and augmented ($n=1524$) datasets for distribution validation.

3.4. Hyperparameter Optimization Strategy

3.4.1. AdaBoost

The AdaBoost classifier was implemented using the ada package, and the Gentle AdaBoost variant (`type = "gentle"`) was preferred. The boosting iteration count was set to `iter = 100`. This value represents a parameter range commonly used in the literature that provides a balance between performance and complexity. Higher iteration counts are known to contribute only slightly to accuracy improvement but increase model complexity.

The growth of the decision trees used as weak learners was regulated with the `rpart.control()` function. The maximum tree depth was set to `maxdepth = 14`; this value was adjusted to limit excessive complexity while providing sufficient decomposition power appropriate to the data size and feature count. The complexity parameter was set to `cp = -1`, and internal cross-validation (`xval = 0`) was disabled. Thanks to this choice, the machine learning model was able to learn the optimal limits independently of early pruning.

3.4.2. *k*-Nearest Neighbour

In this study, the *k*-Nearest Neighbor (kNN) algorithm, another machine learning method used, was optimized for the number of neighbors (*k*) parameter. The *k* value was systematically tested in the range of 1–100, and the model was trained for each value, with accuracy calculated on an independent test dataset. The obtained accuracy values were compared, and the *k* value providing the highest performance was determined as the optimal parameter. The final kNN model was reconstructed with this optimal *k* value, and its performance was reported via the confusion matrix. This ensured a balanced choice between the risks of overfitting that can be seen with small *k* values and overgeneralization that can occur with large *k* values.

3.4.3. Naive Bayes (with Laplace optimization)

The Laplace correction parameter for the Naive Bayes model was systematically optimized to mitigate the zero-probability problem and improve classification performance. Model accuracy was measured for each adjustment, with Laplace values evaluated on a range of 1–30. This method is designed to evaluate the effect of the correction on prediction stability, especially in datasets characterized by unbalanced class distributions or low-frequency feature combinations. The Laplace value that yielded the highest accuracy was selected as the optimal parameter for the final model.

3.4.4. Random Forest (RF)

The Random Forest machine learning model consists of 500 decision trees (*ntree* = 500). Literature indicates that a tree count between 300 and 1000 is generally sufficient for performance stability. A 500-tree structure was chosen to keep computational costs at a reasonable level and reduce variance. The '*importance* = TRUE' parameter was enabled for variable significance analysis. The proximity matrix between samples was obtained using the '*proximity* = TRUE' option.

The '*mtry*' parameter is left at its default value. Using

$$mtry \approx \sqrt{p}$$

as the default in classification problems theoretically increases intertree diversity. Since further testing on '*mtry*' did not significantly improve performance in preliminary analyses, the default structure has been retained for the simplicity and reproducibility of the model.

3.4.5. Support Vector Machines (SVM)

The Support Vector Machine (SVM) model was run using a radial basis function (RBF) kernel. To optimize model performance, a systematic grid search was applied to two key hyperparameters: the penalty parameter (*cost*) and the kernel parameter (*gamma*).

The *gamma* parameter was scanned in the range of 0–0.1 with increments of 0.005; the *cost* parameter was scanned in the range of 20–25. These ranges were chosen to prevent the decision boundary from being overly

flexible or overly rigid. This chosen approach aims to increase the generalization capacity of the model and ensure that the decision boundary fits the data distribution optimally.

3.5. Performance Metrics

To analyze the classifier's classification efficiency, after the training phase is over, the classifier's prediction accuracy should be measured. The actual performance of the classification model is usually done by making use of the data present in the confusion matrix. A confusion matrix is a matrix of at least 2x2 in size utilized in depicting how much of the predicted value of a classification model created due to machine learning algorithms matches with the actual class value. Once a confusion matrix is developed, the performance of prediction models is compared based on accuracy, sensitivity, precision, F1-score, FDR, MCC and AUC criteria.

3.5.1. Accuracy

It is the number of all correct predictions which includes positive predictions and negative ones over total dataset. Accuracy is a value that ranges from 0 to 1, the lowest possible accuracy and highest, respectively. It can also be computed as (1–error rate).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} = 1 - error\ rate$$

3.5.2. Sensitivity

Sensitivity is the proportion of positive examples correctly identified (TP) against all positives presented in the data (TP + FN). This metric is also called recall or true positive rate (TPR). The highest sensitivity is 1, while the lowest is set at zero.

$$Sensitivity = Recall = \frac{TP}{TP + FN}$$

3.5.3. Precision

Precision indicates how many positive predictions a machine learning model calls positive. In other words, it is the ratio of true positives to all positive predictions. The worst precision is 0, and the best precision is 1. The closer it gets to 0, the more false positive predictions the model makes, and the closer it gets to 1, the more accurate almost all positive predictions are.

$$Precision = \frac{TP}{TP + FP}$$

3.5.4. F1-score

The F1-Score value is a criterion that balances precision and sensitivity values. In the case of unequally distributed datasets, using F1-Score instead of precision will prevent an erroneous model selection. F1-Score is calculated by

taking the harmonic mean of sensitivity and precision values. The reason for using the harmonic mean instead of arithmetic is to consider the extreme cases. In the case of 0 and 1 values, the F1-Score is calculated as 0 with the harmonic mean, while it is incorrectly calculated as 0.5 with the arithmetic mean.

$$F_1 \text{ Score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

3.5.5. FDR

False Discovery Rate (FDR) expresses the proportion of samples that a classification model predicts as positive but are actually false positives. In other words, it shows how many of the model's positive predictions are false positives (FP). It is a critical metric for evaluating the false positive rate, especially in fields such as medical diagnosis.

$$FDR = \frac{FP}{TP + FP}$$

3.5.6. MCC

Matthews Correlation Coefficient (MCC) is a metric that measures model performance in binary classification problems. It is known to provide reliable results, especially in datasets with unbalanced class distributions. MCC measures the correlation between actual and predicted classes, evaluating all confusion matrix elements (TP, TN, FP, FN) together.

$$MCC = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

3.5.7. Area under the curve (AUC)

AUC stands for Area Under the Curve; that is, the area under the ROC curve. It measures a model's ability to discriminate between classes and typically ranges from 0.5 to 1.0. An AUC value of 0.5 for any model means that the model has not established a recognizable discrimination ability. An AUC value between 0.5 and 0.7 indicates that the model has poor discrimination. AUC values between 0.7 and 0.8 indicate an acceptable model, while values between 0.8 and 0.9 indicate excellent discrimination ability. Values above 0.9 indicate exceptional discrimination. In medical studies, AUC values above 0.80

indicate that the model is clinically valid [50].

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

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

$$AUC = \frac{TPR + (1 - FPR)}{2}$$

4. EXPERIMENTAL RESULTS AND DISCUSSIONS

This study presents a detailed analysis and performance comparison of machine learning classifiers used in breast cancer diagnosis. The WDBC dataset used in the study consists of a total of 569 data points. 381 samples (67%) from the original WDBC dataset were used as training data, while 188 samples (33%) formed the test dataset. The test dataset consisted of 188 cell samples, 121 of which were benign and 67 malignant. To ensure an unbiased assessment and prevent data leakage, the original WDBC dataset (n = 569) was first divided into training (67%, n = 381) and test (33%, n = 188) subsets. Data augmentation techniques were applied only to the training set, which was expanded to 1,524 samples, while the test set remained entirely the same. Classifiers were trained using the augmented training data and evaluated on the original test data, which consisted of 188 cell samples containing 121 benign and 67 malignant cases. Figures 3 and 4 show the confusion matrices derived from the original and augmented training scenarios, respectively.

Figures 3 and 4 show the confusion matrices obtained from the original and augmented WDBC datasets, respectively. In the original dataset, SVM achieved the highest classification performance with 66 true positives, 120 true negatives, only 1 false positive, and 1 false negative. After data augmentation, SVM further improved its performance by completely eliminating false positives (FP = 0) and retaining only one false negative.

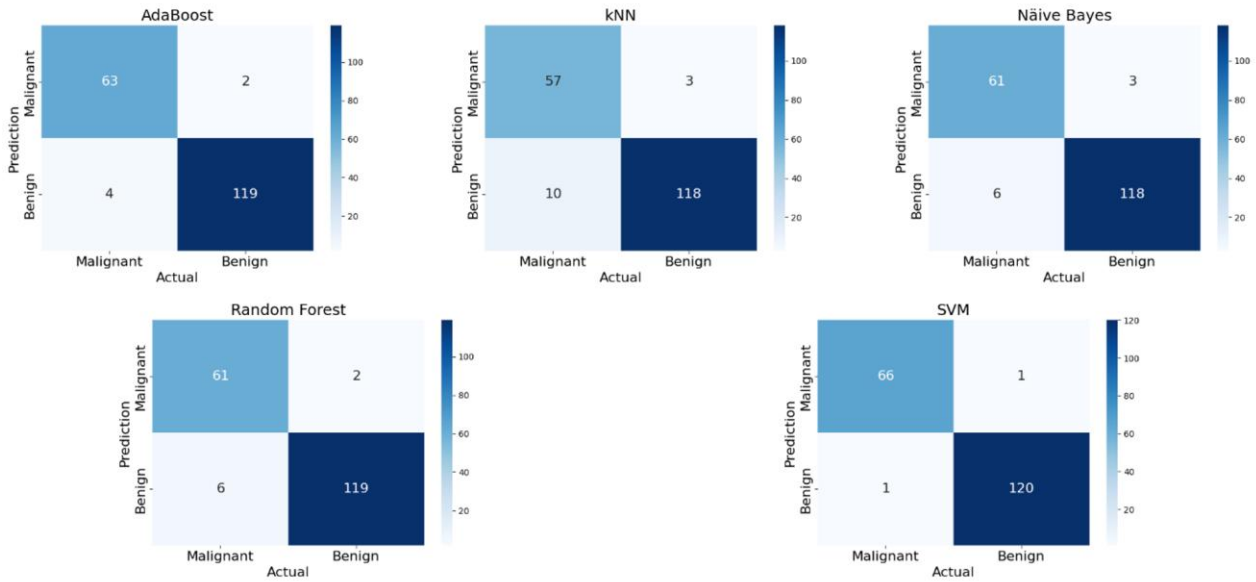


Figure 3. Confusion matrices of five machine learning algorithms based on the WDBC dataset.

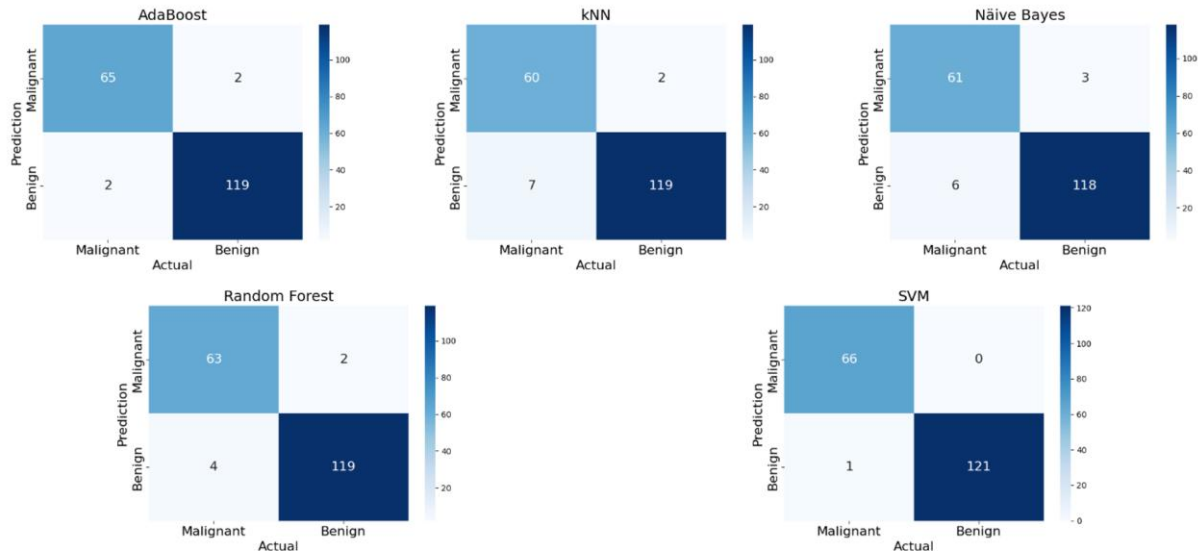


Figure 4. Confusion matrices of five machine learning algorithms based on the augmented dataset.

The AdaBoost and Random Forest algorithms also showed improvement in true positive classifications after data augmentation, while kNN showed a moderate improvement in sensitivity. Naive Bayes performance remained stable in both datasets. Overall, the data augmentation strategy contributed to a reduction in misclassification rates, especially for clinically more critical malignant cases.

Table 2. Performance of models trained with the original dataset on a test dataset.

Metrics	AdaBoost	kNN	Naïve Bayes	RF	SVM
Accuracy	0.9681	0.9309	0.9521	0.9574	0.9894
Precision	0.9692	0.9500	0.9531	0.9683	0.9851
Sensitivity	0.9403	0.8507	0.9104	0.9104	0.9851
F1-Score	0.9545	0.8976	0.9313	0.9385	0.9851
FDR	0.0308	0.0500	0.0469	0.0317	0.0149
MCC	0.9302	0.8486	0.8951	0.9070	0.9768
AUC	0.9619	0.9130	0.9428	0.9470	0.9884

The performance metrics listed in Table 2 represent the baseline classification results on the original WDBC dataset, which were originally presented in [29] and recalculated here focusing on malignant cases as the target class to align with clinical standards. Table 2 shows the classification performance of five machine learning algorithms trained with 381 training data points from the

original WDBC dataset, on 188 test data points. Among all classifiers, SVM yielded the highest overall performance with 98.94% accuracy, 98.51% sensitivity, and the highest MCC value (0.9768). These results for SVM demonstrate a strong correlation between predicted and actual classes. The low FDR (0.0149) confirms that the false positive prediction rate is minimal. AdaBoost ranked second with 96.81% accuracy. kNN showed a relatively lower sensitivity of 85.07%, performing poorer in detecting malignant cases. Naive Bayes produced moderate results across all metrics. Overall, the results show that SVM has the most reliable classification performance on the original dataset, particularly in terms of robustness (MCC) and discrimination ability (AUC = 0.9884).

Table 3. Performance of models trained with an augmented dataset on a test dataset.

Metrics	AdaBoost	kNN	Näive Bayes	RF	SVM
Accuracy	0.9787	0.9521	0.9521	0.9681	0.9947
Precision	0.9701	0.9677	0.9531	0.9692	1.0000
Sensitivity	0.9701	0.8955	0.9104	0.9403	0.9851
F1-Score	0.9701	0.9302	0.9313	0.9545	0.9925
FDR	0.0299	0.0323	0.0469	0.0308	0.0000
MCC	0.9536	0.8954	0.8951	0.9302	0.9884
AUC	0.9768	0.9395	0.9428	0.9619	0.9925

Table 3 shows the classification performance of five machine learning algorithms trained with 1524 augmented training data obtained by applying data augmentation techniques to 381 training data points of the original WDBC dataset, on 188 test data points. SVM achieved the highest accuracy (99.47%), sensitivity (0.9851), and zero false discovery rate (FDR = 0.0000), demonstrating the complete elimination of false positive predictions. The MCC (0.9884) and AUC (0.9925) values of SVM show improved discrimination after augmentation. AdaBoost also showed a significant improvement, increasing its accuracy to 97.87%. Random Forest showed an improvement in sensitivity (0.9403) and provided better malignant tumor detection after augmentation. Naive Bayes remained largely unchanged and showed limited sensitivity to augmentation techniques. Overall, the results demonstrate that the proposed data augmentation strategy, especially for SVM, improves classifier generalization and reduces false classification rates.

Table 4. Comparison of original and augmented WDBC dataset results with studies in the literature.

Author (s)	Method	Accuracy
Cabanillas-Carbonell [30]	AdaBoost	0.9700
Zhao et al. [31]	SVM	0.9860
Alzboon et al. [32]	SVM	0.9750
Jakhar et al. [33]	AdaBoost	0.9724
Chen et al. [34]	RF	0.9650
Ozcan et al. [35]	AdaBoost	0.9806
Mashudi et al. [36]	AdaBoost	0.9877
Imran et al. [37]	AdaBoost	0.9824
Prastyo et al. [38]	kNN	0.9649
Aamir et al. [39]	RF	0.9807
Nuzhat et al. [40]	SVM + RF + GBM	0.9808
Tawil et al. [42]	LightGBM	0.9800
Mezagharni et al. [43]	CMGWO+RF	0.9912
Salhi et al. [44]	MLP+BBPSO	0.9600
Zhu et al. [45]	LightGBM-PSO	0.9900
This study WDBC dataset	SVM	0.9894
This study augmented WDBC dataset	SVM	0.9947

Cabanillas-Carbonell and Zapata-Paulini compared AdaBoost, NB, and RF algorithms. The accuracy rates of the algorithms were 0.97, 0.91, and 0.97, respectively. In another study by Zhao et al., the accuracy rates reported for SVM and RF methods were 0.986 and 0.965. In the analysis of Alzboon et al., the accuracy rates of AdaBoost, kNN, NB, RF, and SVM methods were found to be 0.914, 0.807, 0.935, 0.949, and 0.975, respectively. Similarly, SVM and RF algorithms have high accuracy rates in other studies. The SVM accuracy rate obtained in this study is 0.9894, one of the highest values in literature. In conclusion, the SVM algorithm demonstrates the best performance in breast cancer diagnosis, making it a promising candidate for decision support systems.

Several machine learning models were trained on the augmented dataset. The SVM algorithm achieved the highest accuracy rate with 99.47%. This was followed by AdaBoost (97.87%) and RandomForest (96.81%). Compared to the original dataset, there is a general improvement in the results obtained using data augmentation techniques. In particular, the performance of the SVM algorithm improved from 98.94% accuracy on the original dataset to 99.47% on the augmented dataset. Similarly, the AdaBoost algorithm showed better results with data augmentation techniques.

5. LIMITATIONS OF THE STUDY

This study has some limitations. The WDBC dataset originally contained 569 instances, and although this number was increased using data augmentation techniques, it is still limited compared to real-world data. Data augmentation techniques have helped to reduce the imbalance between the 357 benign and 212 malignant samples in the original dataset, but the distribution between classes is still not exactly equal and there is still a

risk that models are biased towards the majority class. Furthermore, artificial samples generated by data augmentation techniques may not be fully representative of real patient data. Although data augmentation techniques provide significant improvements, larger and more diverse data sets are needed to evaluate the performance of models in real clinical settings.

6. CONCLUSION

In this study, we focus on the comparative results of five different machine learning algorithms (AdaBoost, k-NN, NB, RF and SVM) for breast cancer diagnosis. Experiments on the Wisconsin Diagnostic Breast Cancer (WDBC) dataset and the results obtained using data augmentation techniques revealed that the SVM algorithm outperformed the others. The SVM algorithm achieved the highest success with data augmentation techniques, reaching 99.47% accuracy. The AdaBoost algorithm showed the second-best performance with 97.87% accuracy. Random Forest algorithm ranked third with 96.81% accuracy. The data augmentation techniques significantly improved the results on the original dataset. The accuracy of the SVM algorithm increased from 98.94% to 99.47%. Similarly, the AdaBoost algorithm also showed better results with data augmentation techniques. These results show that data augmentation techniques are an effective method to improve the performance of machine learning models used in breast cancer diagnosis and the SVM algorithm is a promising candidate for preliminary diagnostic aid.

Declaration of Ethical Standards

The article does not contain any studies with human or animal subjects.

Credit Authorship Contribution Statement

All authors contributed to the design and execution of the study. Ayça Acet was responsible for the conceptualization of the study, the application of machine learning models, the data augmentation process, data preparation, statistical analysis, and the writing of the article. Abdullah Erhan Akkaya supervised the research process, contributed to the development and validation of the methodology, ensured the critical review of the intellectual content, and supported the overall structure and scientific rigor of the study. All authors contributed to the interpretation of the findings, critically reviewed the article, and approved the final version of the article.

Declaration of Competing Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Funding / Acknowledgements

No funding or research grants were received during the

preparation of this study.

Data Availability

The dataset can be downloaded from <https://archive.ics.uci.edu/dataset/17/breast+cancer+wisconsin+diagnostic>.

References

- [1] C. Fang *et al.*, "Regulatory Mechanisms of O6-Methylguanine Methyltransferase Expression in Glioma Cells," *Sci. Prog.*, vol. 108, no. 2, 2025, doi: 10.1177/00368504251345014.
- [2] S. Zhu, N. Zhou, Q. Li, and X. Liu, "Rewiring Immune Suppression in NSCLC: Roles and Plasticity of Tregs and Th17 Cells," *Front. Immunol.*, vol. 16, 2025, doi: 10.3389/fimmu.2025.1658848.
- [3] T. Zhu, X. Teng, Q. Jiao, and Y. Ren, "T-cell Exhaustion From a Multiomics Perspective: Differentiation Mechanisms and Regulatory Networks in the Journey From Progenitor-Exhausted T cells to Terminally Exhausted T Cells," *Clin. Transl. Med.*, vol. 16, no. 2, 2026, doi: 10.1002/ctm2.70609.
- [4] Z. Lv *et al.*, "The Dual Immunomodulatory Role of B Cells in Tumorigenesis: Mechanisms, Microenvironment Crosstalk, and Therapeutic Implications," *Front. Immunol.*, vol. 16, 2025, doi: 10.3389/fimmu.2025.1649812.
- [5] F. P. Fabrizio, "Unlocking Lung Cancer Cell Dormancy: An Epigenetic Perspective," *Int. J. Mol. Sci.*, vol. 26, no. 22, p. 10997, 2025, doi: 10.3390/ijms262210997.
- [6] M. Kos *et al.*, "From Defense to Disease: How the Immune System Fuels Epithelial-Mesenchymal Transition in Ovarian Cancer," *Int. J. Mol. Sci.*, vol. 26, no. 9, p. 4041, 2025, doi: 10.3390/ijms26094041.
- [7] Y. Wang, A. Lin, Z. Liu, Q. Cheng, J. Zhang, and P. Luo, "Tumor Microenvironment Onmyoji: Cytokines With Dual Protumor and Antitumor Roles," *Cancer Commun.*, vol. 46, 2026, doi: 10.34133/cancomm.0008.
- [8] M. Ma and D. Q. Li, "Autophagy-Related Proteins in Triple-Negative Breast Cancer: From Molecular Insights to Therapeutic Applications," *Int. J. Mol. Sci.*, vol. 26, no. 18, p. 9231, 2025, doi: 10.3390/ijms26189231.
- [9] X. Huang, J. He, H. Sun, Y. Wu, R. Gu, and Z. Li, "PKM2-driven Metabolic Reprogramming in Digestive System Tumors: Mechanisms, Therapeutic Advances, and Clinical Challenges," *Front. Immunol.*, vol. 16, 2025, doi: 10.3389/fimmu.2025.1634786.
- [10] L. A. Torland *et al.*, "Benign breast tumors may arise on different immunological backgrounds," *Mol. Oncol.*, vol. 18, no. 10, pp. 2495-2509, 2024.
- [11] A. Sucharitha and D. Bhuvana, "An overview of cancer," *Adv. Canc. Chemther. Pharmacol.*, vol. 1, no. 1, pp. 1-8, 2023.
- [12] X. Mao *et al.*, "Association of reproductive risk factors and breast cancer molecular subtypes: a systematic review and meta-analysis," *BMC Cancer* 2023 23:1, vol. 23, no. 1, 2023-07-10, doi: 10.1186/s12885-023-11049-0.
- [13] E. I. Obeagu and G. U. Obeagu, "Breast cancer: A review of risk factors and diagnosis," *Medicine*, vol. 103, no. 3, January 19, 2024, doi: 10.1097/MD.00000000000036905.
- [14] J. D. Figueroa *et al.*, "Reproductive factors and risk of breast cancer by tumour subtypes among Ghanaian women: A population-based case-control study," *Int. J. Cancer*, vol. 147, no. 6, 2020/09/09, doi: 10.1002/ijc.32929.
- [15] A. Bothou *et al.*, "Breastfeeding and Breast Cancer Risk: Our Experience and Mini-review of the Literature," *Mater. Sociomed.*, vol. 34, no. 1, 2022/03, doi: 10.5455/msm.2022.33.28-32.
- [16] Z. Wang *et al.*, "CTHRC1 is a Potential Prognostic Biomarker and Correlated with Macrophage Infiltration in Breast Cancer," *Int. J. Gen. Med.*, vol. 15, 2022, doi: 10.2147/IJGM.S366272.
- [17] M. Bieuville *et al.*, "Frontiers | Number of lifetime menses increases breast cancer occurrence in postmenopausal women at high familial risk," *Front. Ecol. Evol.*, vol. 11, 2023/02/02, doi: 10.3389/fevo.2023.912083.
- [18] L. Odén *et al.*, "Plasma osteoprotegerin and breast cancer risk

- in BRCA1 and BRCA2 mutation carriers," *Oncotarget*, vol. 7, no. 52, 2016/12/12, doi: 10.18632/oncotarget.13417.
- [19] S. Nag *et al.*, "Risk factors for the development of triple-negative breast cancer versus non-triple-negative breast cancer: a case-control study," *Sci. Rep.*, vol. 13, no. 1, p. 13551, 2023.
- [20] F. A. C. d. Luz *et al.*, "Frontiers | Survival differences between women and men in the non-reproductive cancers: Results from a matched analysis of the surveillance, epidemiology, and end-results program," *Front. Public Health*, vol. 10, 2023/01/06, doi: 10.3389/fpubh.2022.1076682.
- [21] H. Rumgay *et al.*, "Global burden of cancer in 2020 attributable to alcohol consumption: a population-based study," *Lancet Oncol.*, vol. 22, no. 8, 2021/08/01, doi: 10.1016/S1470-2045(21)00279-5.
- [22] F. Bray *et al.*, "Global cancer statistics 2022: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries," *CA Cancer J Clin*, vol. 74, no. 3, pp. 229-263, 2024, doi: <https://doi.org/10.3322/caac.21834>.
- [23] L. Wilkinson and T. Gathani, "Understanding breast cancer as a global health concern," *Br J Radiol*, vol. 95, no. 1130, p. 20211033, 2022.
- [24] J. Ferlay *et al.* "Global Cancer Observatory: Cancer Today." Lyon, France: International Agency for Research on Cancer. <https://gco.iarc.who.int/today> (accessed 19/08/2024, 2024).
- [25] I. Guvenc, G. Guvenc, S. Tastan, and A. Akuz, "Identifying Women's Knowledge about Risk Factors of Breast Cancer and Reasons for Having Mammography," *Asian Pac J Cancer Prev*, vol. 13, no. 8, 2012, doi: 10.7314/APJCP.2012.13.8.4191.
- [26] T. Anuk and H. Çantay, "Do women living in northeast Anatolia get mammography screening? A hospital-focused cross-sectional study," *J Surg Med*, vol. 6, no. 3, 2022/03/01, doi: 10.28982/josam.982615.
- [27] R. Ali, A. England, M. F. McEntee, C. Mercer, A. Tootell, and P. Hogg, "Effective lifetime radiation risk for a number of national mammography screening programmes," *Radiography*, vol. 24, no. 3, pp. 240-246, 2018/08/01, doi: 10.1016/j.radi.2018.02.001.
- [28] M. E. Akin, "Rate and reasons of missed screening mammography in the COVID-19 pandemic from Turkey," *J Health Sci Med*, vol. 5, no. 4, 2022-07-20, doi: 10.32322/jhsm.1110424.
- [29] A. Acet, "Prediction of breast cancer using SVM, NB, KNN, AdaBoost and Random Forest classification algorithms," Master Thesis, Computer Engineering, İnönü University, Malatya, Türkiye, Institute of Science and Technology, 2022. [Online]. Available: <https://tez.yok.gov.tr/UlusalTezMerkezi>
- [30] M. Cabanillas-Carbonell and J. Zapata-Paulini, "Improving the Accuracy of Oncology Diagnosis: A Machine Learning-Based Approach to Cancer Prediction," *Int. J. Online Biomed. Eng.*, vol. 20, no. 11, 2024/08/08, doi: 10.3991/ijoe.v20i11.49139.
- [31] J. Zhao, T. Y. Pan, W. Yau, H. Lu, and Z. Liu, "Analysis of classification algorithms: Insights from MNIST and WDBC datasets," *Applied and Computational Engineering*, vol. 79, no. 1, 2024/07/25, doi: 10.54254/2755-2721/79/20241622.
- [32] M. S. Alzboon, S. Qawasmeh, M. Alqaraleh, A. Abuashour, A. F. Bader, and M. Al-Batah, "Machine Learning Classification Algorithms for Accurate Breast Cancer Diagnosis | IEEE Conference Publication | IEEE Xplore," 2023 3rd International Conference on Emerging Smart Technologies and Applications (eSmarTA), 2023.
- [33] A. K. Jakhar, A. Gupta, M. Singh, A. K. Jakhar, A. Gupta, and M. Singh, "SELF: a stacked-based ensemble learning framework for breast cancer classification," *Evol. Intell.*, vol. 17, no. 3, 2023-01-31, doi: 10.1007/s12065-023-00824-4.
- [34] H. Chen, N. Wang, X. Du, K. Mei, Y. Zhou, and G. Cai, "Classification Prediction of Breast Cancer Based on Machine Learning," *Comput. Intell. Neurosci.*, vol. 2023, pp. 1-9, 2023, doi: 10.1155/2023/6530719.
- [35] I. Ozcan, H. Aydin, and A. Cetinkaya, "Comparison of Classification Success Rates of Different Machine Learning Algorithms in the Diagnosis of Breast Cancer," *Asian Pac J Cancer Prev*, vol. 23, no. 10, 2022/10, doi: 10.31557/APJCP.2022.23.10.3287.
- [36] N. A. Mashudi, S. A. Rossli, N. Ahmad, and N. M. Noor, "Comparison on Some Machine Learning Techniques in Breast Cancer Classification | IEEE Conference Publication | IEEE Xplore," 2020 IEEE-EMBS Conference on Biomedical Engineering and Sciences (IECBES), 2021, doi: 10.1109/IECBES48179.2021.9398837.
- [37] B. Imran, H. Hambali, A. Subki, Z. Zaeniah, A. Yani, and M. R. Alfian, "Data Mining Using Random Forest, Naïve Bayes, and Adaboost Models for Prediction and Classification of Benign And Malignant Breast Cancer," *J. Pilar Nusa Mandiri*, vol. 18, no. 1, 2022/03/09, doi: 10.33480/pilar.v18i1.2912.
- [38] P. H. Prastyo, I. G. Y. Paramartha, M. S. M. Pakpahan, and I. Ardiyanto, "Predicting Breast Cancer: A Comparative Analysis of Machine Learning Algorithms," *Proceeding International Conference on Science and Engineering*, vol. 3, 2020/04/30, doi: 10.14421/icse.v3.545.
- [39] S. Aamir *et al.*, "Predicting Breast Cancer Leveraging Supervised Machine Learning Techniques," (in English), *Comput Math Method M*, vol. 2022, Aug 16 2022, doi: Artn 5869529, 10.1155/2022/5869529.
- [40] N. Nuzhat, F. Islam, A. S. Sikder, and N. R. Chakraborty, "Comparative Performance Analysis of Ensemble Models for Breast Cancer Classification," *International Journal of Imminent Science & Technology*, vol. 2, no. 2, 2024, doi: 10.70774/ijist.v2i2.27.
- [41] Y. Pristyanto and D. Wirantanu, "Comparison of Genetic Algorithm and Recursive Feature Elimination on High Dimensional Data," *Jurnal Resti (Rekayasa Sistem Dan Teknologi Informasi)*, vol. 8, no. 2, pp. 189-198, 2024, doi: 10.29207/resti.v8i2.5375.
- [42] A. A. Tawil, L. Almazaydeh, B. Alqudah, A. Z. Abualkashik, and A. A. Alwan, "Predictive Modeling for Breast Cancer Based on Machine Learning Algorithms and Features Selection Methods," *International Journal of Electrical and Computer Engineering (Ijece)*, vol. 14, no. 2, p. 1937, 2024, doi: 10.11591/ijece.v14i2.pp1937-1947.
- [43] A. Mezaghrani, M. Deakla, and K. Djemal, "Novel Feature Selection Method for Accurate Breast Cancer Classification Using Correlation Coefficient and Modified GWO Algorithm," *Computer Science Journal of Moldova*, vol. 32, no. 2(95), pp. 175-198, 2024, doi: 10.56415/csjm.v32.10.
- [44] A. Salhi, R. Alshamrani, A. Althbiti, A. Ismail, M. AbdelRahman, and B. M. Hassan, "Optimizing High Dimensional Data Classification With a Hybrid AI Driven Feature Selection Framework and Machine Learning Schema," *Scientific Reports*, vol. 15, no. 1, 2025, doi: 10.1038/s41598-025-08699-4.
- [45] J. Zhu, B. Yin, C. Wu, C. Yin, R. Chen, and Y. Ding, "An Integrated Approach of Feature Selection and Machine Learning for Early Detection of Breast Cancer," *Scientific Reports*, vol. 15, no. 1, 2025, doi: 10.1038/s41598-025-97685-x.
- [46] O. F. Ayepeku, "Analysis and Visualization of Breast Cancer Prediction Through Machine Learning Models," *Sistemasi*, vol. 13, no. 3, p. 1178, 2024, doi: 10.32520/stmsi.v13i3.4100.
- [47] R. Ihaka and R. Gentleman, "R: A language for data analysis and graphics," *J. Comput. Graphical Stat.*, vol. 5, no. 3, p. 299, 1996.
- [48] L. Tierney, "The R Statistical Computing Environment," *Lect. Notes Stat.*, 2012, doi: 10.1007/978-1-4614-3520-4_41.
- [49] N. Vismam and K. S. Rao, "R Programming in Different Fields," *Int. J. Comput. Sci. Eng. Technol.*, vol. 9, pp. 1-8, 2019.
- [50] K. Smith and S. Climer, "Heterogeneity impacts biomarker discovery for precision medicine," *medRxiv*, 2022-08-01, doi: 10.1101/2022.02.14.22270972.

*Research Article***An Offline-First Mobile Reporting System for Digital One Health Surveillance in Resource-Constrained Settings****Edoghogho Olaye ^{a,*} , Daniel Obuh ^a**^a Department of Computer Engineering, Faculty of Engineering, University of Benin, Edo State, Nigeria

ARTICLE INFO

Article history:

Received 23 May 2026

Accepted 30 June 2026

*Keywords:*One Health surveillance,
mobile health (mHealth),
Design Science Research
Methodology,
offline-first systems,
conversational AI,
public health informatics

ABSTRACT

The increasing interdependence of human, animal, and environmental health highlights the need for integrated One Health surveillance systems capable of supporting timely and coordinated disease reporting. In many low- and middle-income settings, existing surveillance workflows remain fragmented, paper-based, and poorly suited for field conditions. This study employs the Design Science Research Methodology (DSRM) to design, develop, and evaluate a mobile One Health surveillance application that addresses these limitations. The proposed solution is a cross-platform, offline-capable mobile system supporting structured digital reporting, local data storage, and an AI-assisted conversational reporting interface aimed at reducing reporting burden. The artifact was demonstrated through simulated real-world usage scenarios and evaluated using functional testing and usability assessment. Results show a System Usability Scale (SUS) score of 78.4, indicating good overall usability, while the conversational interface reduced average reporting time despite a higher learning curve. The findings demonstrate the feasibility of offline-first mobile surveillance systems for improving One Health data capture in resource-constrained environments.

This is an open access article under the CC BY-SA 4.0 license.
(<https://creativecommons.org/licenses/by-sa/4.0/>)

1. INTRODUCTION

The adoption of mobile devices or smartphones has skyrocketed in recent years. Nigeria's smartphone user base has grown to roughly 50 million, reflecting one of the world's most dynamic mobile markets. Despite this rapid smartphone growth, basic phones remain the most widely used type of handset in the nation, mirroring a broader continental pattern [1]. Concurrently, the healthcare sector has increasingly leveraged mobile applications for critical tasks such as remote patient monitoring, telemedicine, and public health surveillance. These technologies have been shown to improve healthcare access, patient engagement, and the real-time tracking of diseases [2]. This established success in human-centric health models presents a compelling precedent for adapting mobile app solutions to the interconnected challenges of One Health, where monitoring the health of humans, animals, and environments is essential.

The One Health concept is founded on the understanding that the health of humans, animals, and their environments are profoundly interconnected [3]. Existing mobile health tools have limitations, including complex reporting forms that reduce usability and hinder efficient data entry.

The purpose of this study is to solve the problem of reporting One Health cases to DOHS using the Design Science Research Methodology (DSRM).

2. RELATED WORKS

The increasing adoption of mobile health (mHealth) technologies has significantly influenced how healthcare systems collect, transmit, and analyze surveillance data. Numerous studies have demonstrated the transformative impact of mobile-based solutions in improving data quality, timeliness, and accessibility across healthcare contexts. [4] conducted a foundational systematic review

* Corresponding author. E-mail address: edoghogho.olaye@uniben.edu
DOI: 10.58190/ijamec.2026.172

which showed that mobile applications enhance healthcare service delivery by enabling rapid disease reporting, patient monitoring, and improved communication between health workers and public health systems. Similarly, [5] highlighted the effectiveness of SMS and mobile platforms in infectious disease surveillance across low- and middle-income countries, demonstrating their suitability for resource-constrained environments. These findings align with [6], who emphasized that mobile health tools improve reach and engagement but face limitations such as low usability—issues that remain prevalent in current surveillance tools.

In the African context, mobile-based solutions have also shown considerable promise in supporting public health surveillance. [7] highlighted that during efforts to combat Ebola outbreaks, mobile surveillance systems played a critical role in enhancing early outbreak detection and improving response coordination.

The One Health framework provides a broader interdisciplinary context for these technological advancements. The concept, as articulated by FAO, WHO, and OIE [8], positions human, animal, and environmental health as interdependent sectors requiring integrated surveillance approaches. However, several scholars note that current surveillance mechanisms still operate in sectoral silos. [9] argue that disjointed reporting structures create blind spots, weakening cross-species disease detection and limiting timely response. [10] emphasize that despite increasing recognition of the One Health paradigm, the operationalization of integrated surveillance remains constrained by technological, institutional, and data-sharing challenges.

To address these concerns, several mobile-based surveillance innovations have emerged. A notable example is the mobile app and dashboard system developed by [11], which demonstrated that integrating mobile data capture with real-time analytics significantly enhances the accuracy, completeness, and timeliness of outbreak reporting. The system allowed health workers to submit outbreak information from any location using either a smartphone or desktop interface, while the dashboard provided daily, weekly, and historical summaries to support informed public health decision-making. This study illustrates how mobile platforms can strengthen early detection and streamline response workflows. However, despite these advancements, existing solutions still reveal persistent gaps in usability, form complexity, and overall user experience, which continue to hinder widespread adoption, particularly in resource-constrained environments.

Usability remains a persistent challenge across mHealth solutions. [12], in their review of mobile health application usability, noted that many systems fail due to overly complex interfaces and lengthy forms that discourage user engagement. [13] similarly stressed that usability

problems directly affect adoption rates, especially in field environments where health workers require fast, intuitive systems. These limitations directly align with issues identified in existing One Health surveillance tools, where tedious reporting workflows and multi-step forms reduce efficiency and compliance.

Finally, from a methodological standpoint, Design Science Research provides a structured pathway for developing solutions to these challenges. [14] positioned Design Science Research as a rigorous approach for creating innovative artifacts that solve real-world problems. [15] later formalized the Design Science Research Methodology (DSRM) process into six cyclical steps—problem identification, objective definition, design and development, demonstration, evaluation, and communication—making it well-suited for designing complex technology-based interventions such as surveillance applications.

Further, advancements in the development of Flutter-based cross-platform solutions have reinforced its viability for mobile health environments where resources are limited, and research has shown that it is a viable option for cross-platform development, achieving competitive rendering performance – and maintenance overhead – compared to approaches using native dual-codebases [21, 22]. Previously, offline-first and local-first software architectures have been developed and tested, showing that the embedding of synchronization logic on the client side, instead of depending on persistent connectivity, has a significant impact on the completeness of data in intermittent-network situations [23]. Issues related to the security of the mHealth data have also been reaffirmed in recent reviews, suggesting the need for end-to-end encryption, role-based access control, and lightweight authentication mechanisms suitable for the field environment [24, 25].

Collectively, the reviewed literature establishes that while mobile technologies significantly enhance health surveillance, existing systems remain fragmented, sector-specific, and often difficult for field workers to use. No current solution provides a unified, user-friendly mobile platform for integrated human–animal–environment reporting within the Nigerian One Health ecosystem. This gap underscores the need for the mobile surveillance system developed in this study using the Design Science Research Methodology.

3. METHODOLOGY

3.1. Research design

The research method employed is the Design Science Research Methodology (DSRM). DSRM produces knowledge through the creation and evaluation of innovative and useful artifacts aimed at solving real-world challenges [16]. The DSRM framework comprises six

iterative phases: problem identification, definition of solution objectives, design and development, demonstration, evaluation, and communication [17].

3.2. Problem identification

The problem identification phase involved qualitative data collection through semi-structured interviews and direct observations with health surveillance staff at the University of Benin Teaching Hospital. These sessions aimed to identify current challenges in field-based health surveillance reporting and gather functional requirements for the proposed mobile application. The identified user requirements are presented in Table 1.

Table 1. User Requirements

No	User requirement
1	The app must work on both Android and iOS phones without needing two separate development projects.
2	The app must be usable in field conditions with no internet connection (offline capability).
3	Reporting complex health incidents is slow and cumbersome with too many form fields to fill out.
4	The app interface should be simple, intuitive, and easy to navigate for users with varying levels of tech expertise.
5	The app needs to be fast and responsive, not sluggish, when loading data or switching screens.
6	Users require a more efficient method for submitting reports without manually completing every field.

3.3. Define objectives of a solution

Based on the identified requirements, specific solution objectives were defined to guide the development process. Table 2 maps each user requirement to its corresponding technical solution.

Table 2. Requirements-to-Solution Mapping

No	User Requirement	Proposed Solution
1	Cross-platform compatibility	Develop using a Single Codebase with Flutter
2	Offline capability	Implement Local Database (SQLite) to make the app work offline.
3	Tedious, multi-step reporting forms.	Smooth, structured workflow and data handling between each step of the reporting process. Also, dropdowns for selecting some form fields
4	Need for a simple, intuitive interface.	The use of Flutter's rich set of customizable widgets to create clean, responsive, and user-friendly interface.
5	Requirement for a fast, responsive app.	Efficient API calls with modular widget development for fast user experience.
6	Desire for a smarter reporting method.	Conversational UI within the Flutter app that connects to the backend AI agent.

3.4. System design

3.4.1. Use case analysis

Use case diagrams were developed to model system functionality and clarify ambiguities in user interactions [18]. Figure 1 illustrates the primary use cases for field workers (actors), including authentication, location updates, manual report submission, outbreak monitoring, and AI-assisted reporting.

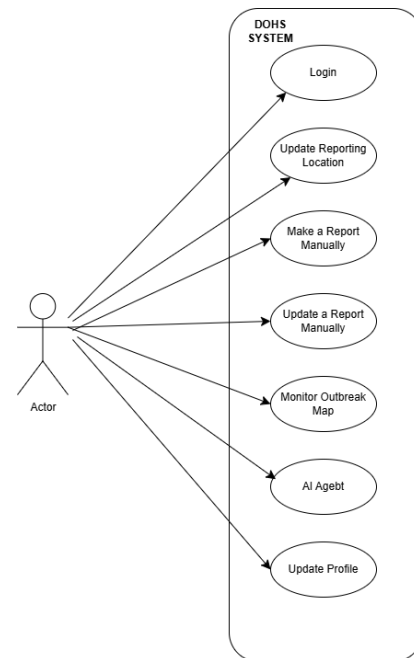


Figure 1. Use Case Diagram

3.4.2. Activity modelling

Activity diagrams were created to represent the sequential and concurrent workflows within the application [19]. Two critical workflows were modeled and presented:

Authentication Workflow: The login process implements secure authentication where users receive credentials via email. Upon entering valid credentials, the system verifies the account against the backend database. Successful authentication redirects users to the dashboard, while failed attempts trigger appropriate error notifications.

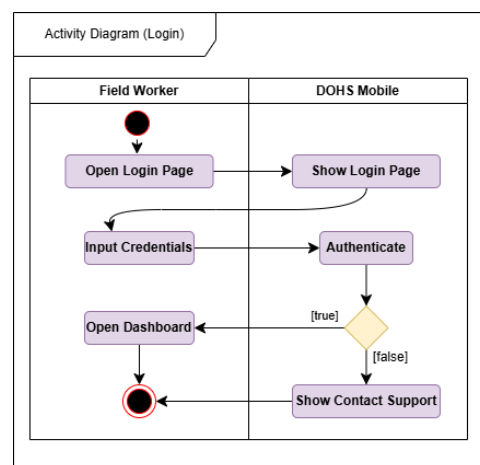


Figure 2. Login Activity Diagram

Manual Report Submission Workflow: This workflow depicts the multi-step process for creating and submitting field reports, including data validation and backend API communication.

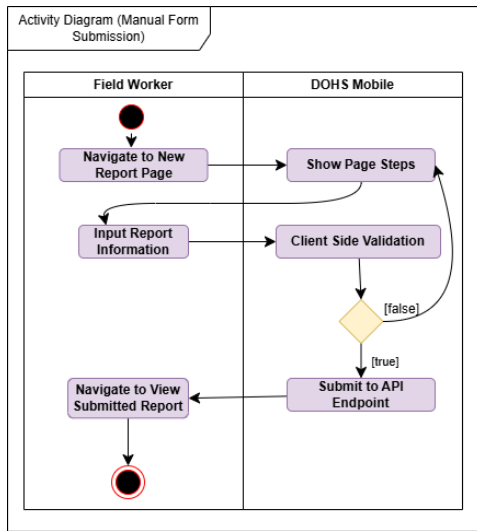


Figure 3. Manual Form Submission Workflow

3.5. Implementation

3.5.1. Development environment

The mobile application was developed using Flutter, leveraging the Dart programming language. Flutter's cross-platform capabilities enabled simultaneous deployment to both Android and iOS platforms from a single codebase. The development process utilized Flutter's package ecosystem (Pub) for dependency management and incorporated Flutter's stateful hot reload feature to accelerate iterative development [20].

3.5.2. Architectural design

The application implements a Provider-centric architecture pattern, establishing unidirectional data flow and clear separation between business logic and user interface components. The architecture consists of three primary layers:

Presentation Layer: Flutter widgets that render the user interface.

Business Logic Layer: Provider classes (e.g., DOHSAuthProvider) utilizing ChangeNotifier for state management.

Data Layer: RESTful API client for backend communication and SQLite for local data persistence.

State changes in Provider objects trigger notifications to registered listeners, resulting in reactive UI updates without manual intervention.

3.5.3. Data exchange protocol

Communication between the mobile client and backend server follows RESTful API principles, with all data exchanges formatted in JSON. This standardized approach ensures efficient serialization, platform independence, and straightforward integration with the backend infrastructure.

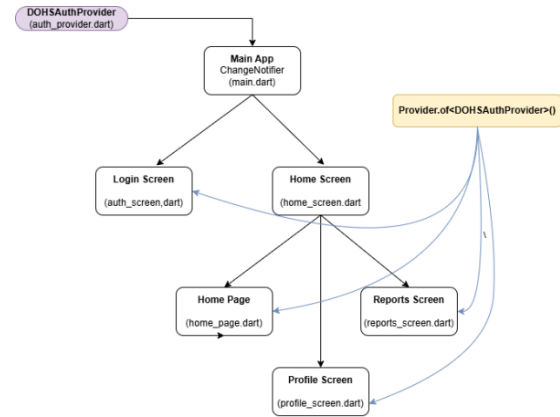


Figure 4. Provider Pattern Workflow for managing Authentication State in Application Lifecycle.

3.5.4. AI Agent Architecture and Prompt Engineering

The AI-assisted conversational reporting module is built on a locally hosted large language model (LLM), specifically DeepSeek-R1:8B, served via Ollama on the backend server. This design eliminates reliance on external API services, preserves data privacy in a health surveillance context, and reduces operational costs typically associated with cloud-based LLM inference.

Core AI logic is implemented directly within the backend server rather than through an external orchestration layer. When a field worker submits a natural language report through the conversational interface, the Flutter client sends the input to the backend via a REST API. The backend then constructs a structured prompt and invokes the locally hosted LLM through the Ollama API. The prompt is engineered to instruct the model to function as a One Health surveillance data extractor, returning structured JSON output containing required fields such as disease name, onset date, confirmation date, symptoms, and outcome status.

Upon receiving the model's response, the backend performs schema validation to ensure that all mandatory fields have been correctly extracted. If any fields are missing, incomplete, or ambiguous, the system initiates a clarification loop by generating targeted follow-up questions, which are sent back to the user through the Flutter interface. This iterative interaction continues until all required fields are sufficiently captured and validated. Once complete, the structured report is processed and stored in the system.

3.6. Evaluation methods

3.6.1. Black-box testing

Functional testing verified that all system components performed according to specifications across both Android and iOS platforms. Test cases covered critical functionalities including authentication, navigation, data submission, and offline operation.

3.6.2. Usability testing

A convenience sample of 25 university students

participated in task-based usability testing. Participants completed structured tasks while metrics were collected on:

- Task completion rates (success/failure).
- Time-on-task (seconds).
- System Usability Scale (SUS) scores.

Participants were selected because they were easily accessible and allowed for rapid evaluation of the system. The university environment allowed for recruitment, testing, and feedback to be done efficiently, multiple times, and in a timely fashion, which helped refine the user interface and get feedback in a controlled manner. This sampling approach, however, makes it difficult to generalize to the larger population of healthcare field workers because the field workers face unique contextual pressures such as time constraints or domain-specific communication. Evaluation with target end-users is therefore still an important goal for future work (as described in Section 5).

4. RESULTS AND DISCUSSION

4.1. System implementation

The development process began with low-fidelity prototyping using Figma, allowing stakeholders to review and provide feedback before full implementation. This approach minimized development rework by addressing design concerns during the planning phase. The final implementation consists of several integrated modules deployed on Android devices for field testing.

4.1.1. Authentication module

The login interface (Figure 5) implements secure credential-based authentication. Field workers access the system using unique usernames and system-generated passwords distributed via secure email channels, ensuring only authorized personnel can submit surveillance data.

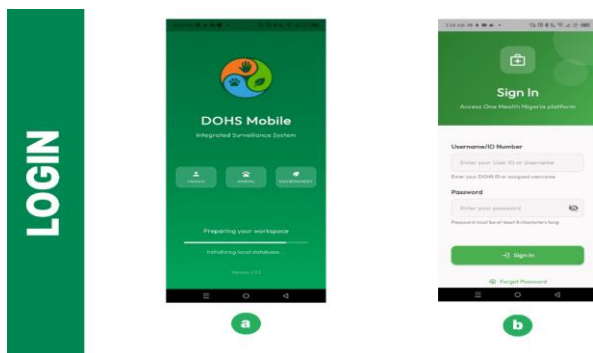


Figure 5. Login Interface

4.1.2. Manual reporting module

The report creation module represents the core functionality of the DOHS application. The traditional reporting interface employs a multi-step wizard approach to guide users through data collection (Figure 6):

Step 1: Disease identification with dropdown menus for standardized disease codes and classification categories.

Step 2: Temporal data entry including dates of onset and confirmation.

Step 3: Clinical details with text areas for symptom descriptions and outcome status selection.

Required field indicators ensure data completeness before submission. This structured approach balances standardized data capture with clinical detail documentation.

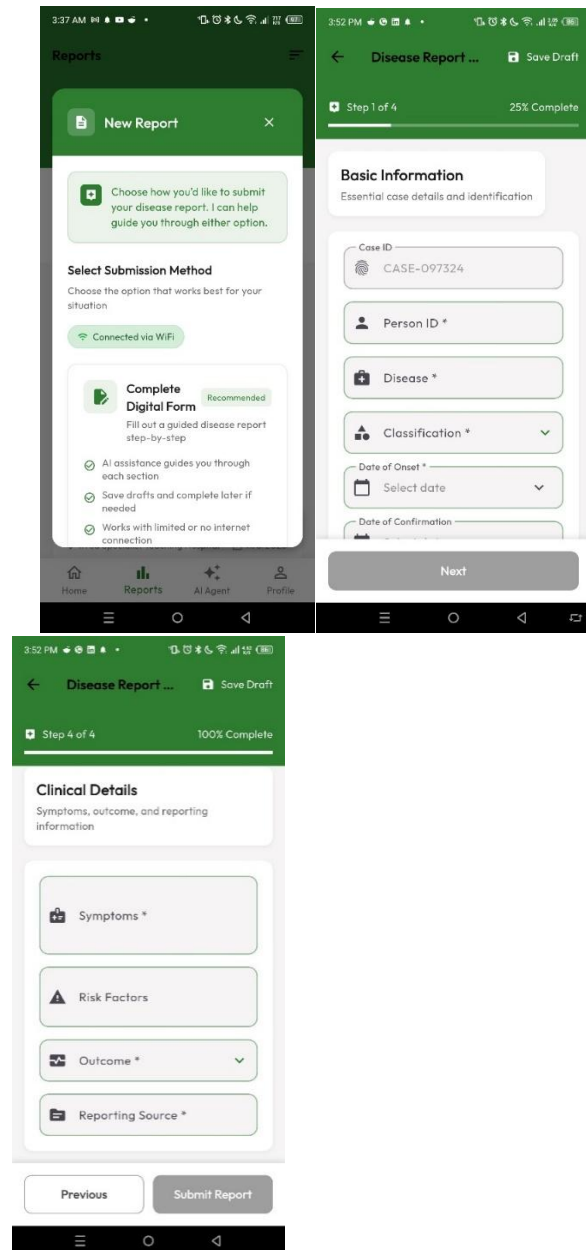


Figure 6. Manual Form Workflow

4.1.3. AI-assisted conversational reporting module

The AI-assisted conversational reporting agent provides an alternative data submission pathway using natural language processing (Figure 7). The chat-style interface reduces the cognitive burden associated with structured forms by allowing field workers to describe cases in natural language.

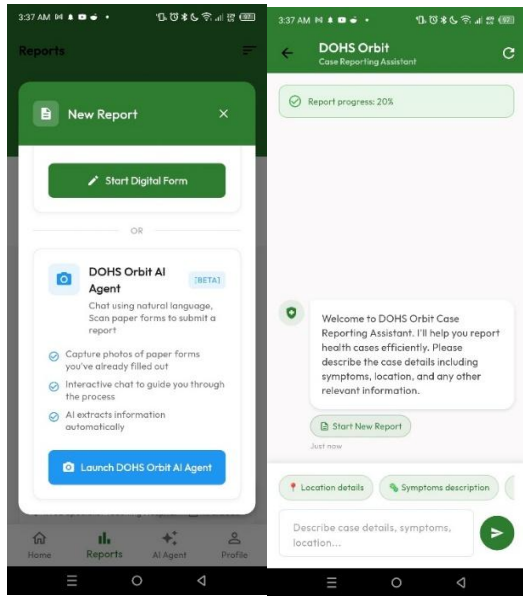


Figure 7. AI Agent for Assisted Reporting Interface.

The AI agent processes user input through the backend server and returns contextual responses (Figure 8):

Success messages: Confirm successful data extraction and report submission as shown in Figure 8.

Clarification requests: Prompt users for missing required fields.

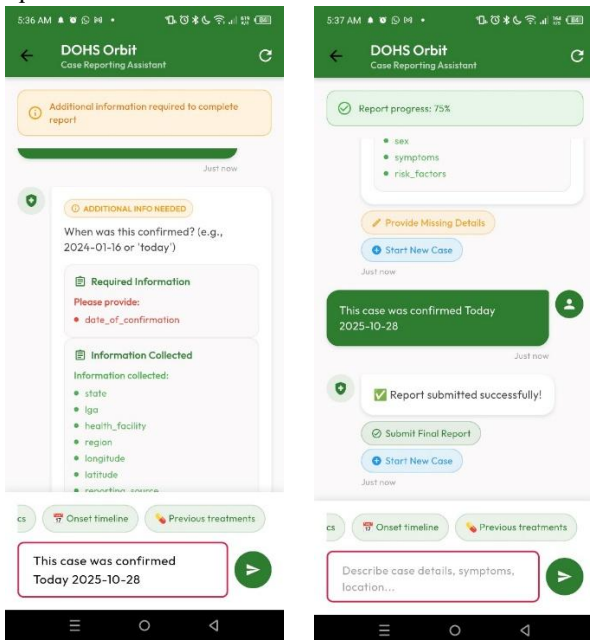


Figure 8. AI reporting agent response message types

4.2. Functional testing results

Black-box testing validated all critical system functionalities across both Android and iOS platforms. Table 3 summarizes the functional test results.

Table 3. Functional testing results

No	Test Case	Result	Status
1	App Launch & Splash Screen	The app opens correctly on both Android and iOS, displaying the DOHS branding and initializing the local database.	Valid
2	User Login & Authentication	The login screen accepts credentials, communicates with the backend, receives a JWT token, and successfully navigates to the Home Dashboard.	Valid
3	Bottom Navigation Bar	Tapping between Home, Reports, and Profile sections correctly switches the main view without reloading the entire app.	Valid
4	Manual Form Reporting	The multi-step form navigates through all steps, validates input, and successfully submits a complete report to the backend (or saves it locally if offline).	Valid
5	AI-assisted conversational reporting	The chat interface sends user messages and displays intelligent replies from the AI agent, eventually resulting in a fully populated and submitted report.	Valid
6	Viewing Report List	The Reports screen correctly displays a scrollable list of the user's past reports, loaded from the local database and/or backend.	Valid

All tested functionalities achieved "Valid" status, indicating the system meets its technical specifications.

4.3. Usability testing results

4.3.1. Task completion analysis

Task-based usability testing with 25 participants revealed differential performance across interface types (Figure 9). Traditional interface tasks demonstrated high success rates:

Manual form submission: 88%

Report retrieval: 100%

Dashboard navigation: 92%

In contrast, the AI-assisted conversational reporting task achieved a 64% success rate, representing a statistically significant difference ($\chi^2 = 4.90$, $p < 0.05$).

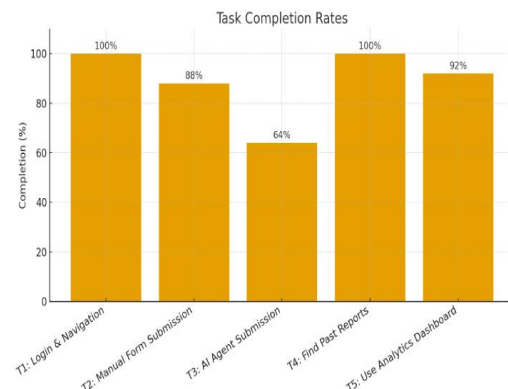


Figure 9. Chart of task completion rates

4.3.2. Efficiency metrics

Time-on-task analysis revealed that successful users completed AI-assisted reporting faster ($M = 132s$, $SD = 47.5s$) compared to manual form submission ($M = 158s$, $SD = 31.2s$). However, the higher standard deviation for AI tasks indicates greater variability in user performance, suggesting an initial learning curve for the conversational interface (Table 4).

Table 4. Task performance metrics

Task	Success Rate (%)	Mean Time-on-Task(s)	Std. Deviation
Manual Form Submission	88	158	31.2
AI Agent Submission	64	132	47.5

The usability results were gained from a student participant sample ($n=25$) and are thus to be regarded in this context. The performance of field-based health workers can differ from what was observed in the field because of domain familiarity and other environmental factors as well as their experience using mobile applications.

4.4. Deployment and knowledge transfer

Following successful evaluation, the application was deployed through multiple channels:

End-user Distribution: The mobile application was made available via the DOHS website for authorized field workers to download and install on their devices.

Documentation: Comprehensive user documentation was developed to explain module functionality and support future system maintenance and feature updates.

Stakeholder Presentation: The completed system was demonstrated to end users and stakeholders, showcasing its functionality and potential impact on One Health surveillance efforts.

Academic Dissemination: Research findings were prepared for journal publication to contribute to the scholarly literature on mobile health surveillance systems and Design Science methodologies.

5. CONCLUSION

This paper utilizes the Design Science Research Methodology to design, develop, and test a mobile One Health surveillance application that helps overcome the challenges of the current paper-based and fragmented reporting processes in the Department of Health Services (DOHS). The system created is based on Flutter, which incorporates structured digital reporting forms, offline data capture, and an AI-assisted conversational reporting feature to improve the reporting efficiency, data quality, and the overall user experience.

System demonstrations established that the application functions as intended in its essential capabilities, and the usability testing yielded a System Usability Scale (SUS)

score of 78.4, indicating good overall usability and acceptance by the users of the application. Though the AI-assisted conversational reporting element showed a steeper learning curve and needs additional improvements, it still showed decreases in reporting time and strong potential for improvement in the future.

In general, the final artifact represents a viable and extensible solution to enhancing real-time One Health information capture and enabling more integrated surveillance in the human, animal, and environmental health sectors. Further enhancements to the conversational reporting agent are planned for future work, along with increased field testing on health professionals and further incorporation of advanced analytics to improve timely detection and response to an outbreak.

Declaration of Ethical Standards

This study did not involve medical, clinical, or invasive procedures. Participant involvement in requirements-gathering and usability evaluation was voluntary, with no sensitive or identifiable personal data collected.

Credit Authorship Contribution Statement

All authors contributed substantially and equally to this study, including conceptualization, methodology, implementation, analysis, and manuscript writing. All authors approved the final manuscript.

Declaration of Competing Interest

The authors declare that there are no conflicts of interest regarding the publication of this manuscript.

Funding / Acknowledgements

This work was supported by the Tertiary Education Trust Fund National Research Fund grant (TETFund Ref: NRF2023/CC/STI/00031/VOL.1).

Data Availability

Data were collected through interviews and usability testing, and are available from the corresponding author upon reasonable request.

References

- [1] A. Owoyemi et al., "Mobile health approaches to disease surveillance in Africa; Wellvis COVID triage tool," *Digital Health*, vol. 7, p. 2055207621996876, 2021.
- [2] A. O. Babatunde et al., "Leveraging mobile health technology towards achieving universal health coverage in Nigeria," *Public Health in Practice*, vol. 2, p. 100120, 2021.
- [3] S. J. Pitt and A. Gunn, "The one health concept," *British Journal of Biomedical Science*, vol. 81, p. 12366, 2024.
- [4] C. Free, G. Phillips, L. Felix, L. Galli, V. Patel, and P. Edwards, "The effectiveness of M-health technologies for improving health and health services: a systematic review protocol," *BMC Res Notes*, vol. 3, no. 1, p. 250, Dec. 2010, doi: 10.1186/1756-0500-3-250.
- [5] C. Dégli, L. S. Suggs, and P. Odermatt, "SMS for disease control in developing countries: a systematic review of mobile health applications," *J Telemed Telecare*, vol. 18, no. 5, pp. 273–281, July 2012, doi: 10.1258/jtt.2012.110810.
- [6] A. B. Labrique, L. Vasudevan, E. Kochi, R. Fabricant, and G.

- Mehl, "mHealth innovations as health system strengthening tools: 12 common applications and a visual framework," *Glob Health Sci Pract*, vol. 1, no. 2, pp. 160–171, Aug. 2013, doi: 10.9745/GHSP-D-13-00031.
- [7] E. Tambo, E. C. Ugwu, and J. Y. Ngogang, "Need of surveillance response systems to combat Ebola outbreaks and other emerging infectious diseases in African countries," *Infect Dis Poverty*, vol. 3, no. 1, p. 29, Aug. 2014, doi: 10.1186/2049-9957-3-29.
- [8] National Framework for One Health. FAO, 2021. doi: 10.4060/cb4072en.
- [9] D. Destoumieux-Garzón et al., "The One Health Concept: 10 Years Old and a Long Road Ahead," *Front. Vet. Sci.*, vol. 5, p. 14, Feb. 2018, doi: 10.3389/fvets.2018.00014.
- [10] A. L. Okello, K. Bardosh, J. Smith, and S. C. Welburn, "One Health: Past Successes and Future Challenges in Three African Contexts," *PLoS Negl Trop Dis*, vol. 8, no. 5, p. e2884, May 2014, doi: 10.1371/journal.pntd.0002884.
- [11] E. Ahn et al., "A Mobile App and Dashboard for Early Detection of Infectious Disease Outbreaks: Development Study," *JMIR Public Health Surveill*, vol. 7, no. 3, p. e14837, Mar. 2021, doi: 10.2196/14837.
- [12] B. C. Zapata, J. L. Fernández-Alemán, A. Idri, and A. Toval, "Empirical Studies on Usability of mHealth Apps: A Systematic Literature Review," *J Med Syst*, vol. 39, no. 2, p. 1, Feb. 2015, doi: 10.1007/s10916-014-0182-2.
- [13] M. Khosravi, S. M. Mojtabaiean, and M. Aghamaleki Sarvestani, "A systematic review on factors influencing Middle Eastern women's utilization of healthcare services: The promise of mHealth," *SAGE Open Medicine*, vol. 12, p. 20503121241276678, Jan. 2024, doi: 10.1177/20503121241276678.
- [14] Hevner, March, Park, and Ram, "Design Science in Information Systems Research," *MIS Quarterly*, vol. 28, no. 1, p. 75, 2004, doi: 10.2307/25148625.
- [15] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, "A Design Science Research Methodology for Information Systems Research," *Journal of Management Information Systems*, vol. 24, no. 3, pp. 45–77, Dec. 2007, doi: 10.2753/MIS0742-1222240302.
- [16] T. Tuunanen, R. Winter, and J. vom Brocke, "Dealing with complexity in design science research: a methodology using design echelons," *MIS quarterly*, vol. 48, no. 2, pp. 427–458, 2024.
- [17] B. Gledson, K. Rogage, A. Thompson, and H. Ponton, "Reporting on the development of a web-based prototype dashboard for construction design managers, achieved through design science research methodology (DSRM)," *Buildings*, vol. 14, no. 2, p. 335, 2024.
- [18] D. K. Kim, "Development of Mobile Cloud Applications using UML," *International Journal of Electrical & Computer Engineering (2088-8708)*, vol. 8, no. 1, 2018.
- [19] T. Ahmad, J. Iqbal, A. Ashraf, D. Truscan, and I. Porres, "Model-based testing using UML activity diagrams: A systematic mapping study," *Computer Science Review*, vol. 33, pp. 98–112, 2019, doi: <https://doi.org/10.1016/j.cosrev.2019.07.001>.
- [20] A. Tashildar, N. Shah, R. Gala, T. Giri, and P. Chavhan, "Application development using flutter," *International Research Journal of Modernization in Engineering Technology and Science*, vol. 2, no. 8, pp. 1262–1266, 2020.
- [21] S. Durai, C. Shyamalakumari and T. Sujithra, "Cloud Computing based Multipurpose E-Service Application using Flutter," 2022 6th International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 2022, pp. 1122–1126, doi: 10.1109/ICCMC53470.2022.9753968.
- [22] A. M. Sattar et al., 'Accelerating cross-platform development with Flutter framework', *J. Open Source Dev.*, vol. 10, no. 2, pp. 1–11, 2023.
- [23] R. Sugumar, 'Federated AI in Offline-First Mobile Health Architectures for Privacy-Preserving Clinical Intelligence', *Int. J. Sci. Res. Technol.*, vol. 8, no. 4, pp. 14589–14600, 2025.
- [24] A. K. I. Riad et al., 'Enhancing hipaa compliance in ai-driven mhealth devices security and privacy', in *2024 IEEE 48th Annual Computers, Software, and Applications Conference (COMPSAC)*, IEEE, 2024, pp. 2430–2435.
- [25] R. Rezaee, M. Khashayar, S. Saeedinezhad, M. Nasiri, and S. Zare, 'Critical criteria and countermeasures for mobile health developers to ensure mobile health privacy and security: mixed methods study', *JMIR MHealth UHealth*, vol. 11, p. e39055, 2023.

*Research Article***Comparative Evaluation of Machine Learning Algorithms for Raisin Variety Classification Based on Morphological Features****Huseyin BULDUK ^{a,*} , Kadir SABANCI ^a** ^a *Electrical and Electronics Engineering, Karamanoglu Mehmetbey University, Karaman, Türkiye*

ARTICLE INFO

Article history:

Received 5 May 2026

Accepted 30 June 2026

*Keywords:*Raisin Classification,
k-Nearest Neighbors (k-NN),
Multilayer Perceptron (MLP),
Support Vector Machines
(SVM),
WEKA

ABSTRACT

Machine learning algorithms enable the rapid, accurate, and automated classification of agricultural products. Compared to human visual assessment, these algorithms provide more consistent and highly accurate determination of product characteristics such as variety and quality. In this study, machine learning algorithms were employed to classify Keçimen and Besni raisin varieties cultivated in Turkey. The Raisin dataset obtained from the UCI Machine Learning Repository was used as the data source. For the classification task, k-Nearest Neighbors (k-NN), Multilayer Perceptron (MLP), and Support Vector Machines (SVM) algorithms were utilized. The classification process was carried out using the WEKA software. Initially, the dataset was divided into 80% training and 20% testing subsets, and classification was performed accordingly. Subsequently, the classification process was repeated using the 10-fold cross-validation method. The obtained performance results were evaluated comparatively among the applied algorithms. The MLP algorithm achieved the highest performance, with an accuracy of 92.78% and an AUC of 0.968 under the 80% training–20% testing scheme. Under 10-fold cross-validation, MLP also produced the highest accuracy and AUC values of 87.22% and 0.928, respectively. Although the k-NN and SVM algorithms also produced satisfactory results, they did not reach the performance level of MLP. The findings demonstrate that the evaluated machine learning algorithms can provide effective classification performance for the two raisin varieties under the experimental conditions considered in this study.

This is an open access article under the CC BY-SA 4.0 license.
(<https://creativecommons.org/licenses/by-sa/4.0/>)

1. INTRODUCTION

The agricultural sector holds significant economic and social importance. Quality control and classification of agricultural products are critical for increasing their market value and meeting consumer demands. Traditional classification methods, which are time-consuming and prone to errors, are increasingly being replaced by automated systems in modern agriculture. Machine learning and deep learning techniques have revolutionized this field by providing high accuracy and efficiency in the classification of agricultural products [1]. In particular, when combined with image processing techniques, these methods enable rapid and reliable classification based on the physical characteristics of products [2].

Machine learning algorithms such as k-Nearest

Neighbors (k-NN), Support Vector Machines (SVM), and Artificial Neural Networks (ANN) are widely used for the classification of agricultural products. For instance, Sonmez et al. achieved 99.84% success in wheat classification with SVM [3]. Bulduk and Sabanci proposed a hybrid model based on Swin Transformer and EfficientNetV2 for the classification of three corn varieties; the single model achieved 99.37% accuracy on 1050 images, while the hybrid model achieved 100% accuracy, demonstrating high performance [4]. In another study, Bulduk and Sabanci compared different SVM kernel functions for date classification; when using 13 features selected with MRMR, the accuracy increased from 91.79% to 92.07% [5]. The results show that feature selection is important in improving classification

* Corresponding author. E-mail address: huseyinbulduk@kmu.edu.tr
DOI: 10.58190/ijamec.2026.173

performance. The success of these methods in the agricultural sector has not only improved product quality but also reduced dependency on manual labor.

Raisins are a rich source of carbohydrates and a nutritious snack containing antioxidants, potassium, fiber, and iron [6]. Türkiye is one of the world's leading grape- and raisin-producing countries; approximately 30% of raisin varieties are consumed as table raisin varieties, 37% are processed as dried products, 3% are used in wine production, and the remaining 30% are used for other purposes [7]. In recent years, numerous studies have been conducted using machine learning and deep learning techniques for raisin varieties and raisin classification, demonstrating an increasing interest in automated classification approaches. Çınar et al. [8] focused on the automated classification of Keçimen and Besni raisin varieties using machine vision and artificial intelligence techniques. By evaluating morphological features through Logistic Regression, Multilayer Perceptron, and Support Vector Machines (SVM), the researchers determined that the SVM model provided the highest classification performance with an accuracy of 86.44%. Khojastehnazhand and Ramezani [9] developed a machine vision system to classify the quality of bulk raisin samples using texture-based features. Their study demonstrated that combining Gray-Level Run Length Matrix (GLRLM) features with a Support Vector Machine (SVM) algorithm achieved an accuracy of 85.55% in a 6-class scenario, although the performance decreased as the classification structure became more complex. Wang et al. [10] introduced a non-invasive classification method for raisins by integrating a total of 74 hybrid features encompassing morphological, color, and texture data extracted from RGB images. Their experimental results demonstrated that the combination of all feature types yielded the highest performance, with the Linear Discriminant Analysis (LDA) model achieving a 99% accuracy rate, significantly surpassing models based on individual feature sets. Yu et al. [11] developed a hybrid classification framework for raisin quality assessment by integrating color and texture features extracted from RGB and HSI color spaces. Utilizing the Least Squares Support Vector Machine (LSSVM) algorithm, the study achieved a 95% accuracy rate across four quality classes, highlighting that the fusion of color histograms and GLCM-based texture data significantly improves model reliability over individual feature sets. Bisri and Man [12] investigated the impact of sampling techniques, specifically stratified and shuffled sampling, on the performance of machine learning models for raisin classification. Their findings revealed that shuffled sampling outperformed stratified sampling with a top accuracy of 88.11%, demonstrating that the choice of sampling strategy is a critical factor in enhancing the reliability and performance of algorithms like Random Forest. Şahin [13] proposed an efficient classification

approach for raisin grains by leveraging feature selection techniques to enhance model performance while reducing complexity. By identifying the most significant morphological feature through Random Forest-based importance analysis and utilizing it within a Multi-Layer Perceptron (MLP) model, the study achieved a superior accuracy rate of 94.07%. These findings suggest that focusing on a limited set of highly effective variables can lead to more accurate and streamlined classification compared to using all available morphological data. Ünal et al. [14] conducted a comparative analysis of Decision Trees and Random Forest algorithms for the binary classification of raisin varieties using a seven-feature dataset. Their experimental results indicated that the Random Forest model achieved a slightly superior and more stable performance with an accuracy of 85.44%, compared to 85.22% for the Decision Tree model, highlighting the effectiveness of classical machine learning methods for datasets with a limited number of features. Guo et al. [15] proposed a non-destructive method for classifying raisin varieties by combining Near-Infrared (NIR) spectroscopy with pattern recognition algorithms such as SVM, MCNN, and an improved AlexNet. Their experimental results showed that the SVM model, integrated with Principal Component Analysis (PCA), achieved a perfect classification accuracy of 100%, indicating that NIR spectroscopy is a highly effective and reliable approach for the rapid identification of raisin cultivars. Zhou [16] evaluated the performance of XGBoost, SVM, MLP, and Logistic Regression algorithms for classifying raisin varieties, identifying the "Extent" feature as the most critical variable in the process. The experimental results indicated that the SVM model outperformed the other classifiers, achieving a 91% accuracy rate and an AUC of 0.91, which underscores the efficacy of classical machine learning in automated agricultural grading. Dirik [17] introduced a hybrid classification approach by integrating Particle Swarm Optimization (PSO) with Artificial Neural Networks (ANN) to enhance the grading accuracy of raisin grains. While traditional models such as KNN and Random Tree achieved lower performance, the proposed PSO-ANN hybrid reached a perfect accuracy rate of 100%, demonstrating the significant impact of metaheuristic optimization on neural network efficiency in agricultural sorting tasks. Ramdhani et al. [18] investigated the impact of Genetic Algorithm (GA) based feature selection on the classification performance of Support Vector Machines (SVM) using the raisin dataset. Their experimental results demonstrated that the GA-optimized SVM model achieved a superior accuracy of 87.67% and an AUC of 0.930, outperforming other traditional algorithms such as Naive Bayes and Decision Trees by effectively refining the feature set. Dewi et al. [19] conducted a comparative analysis of Decision Tree and Support Vector Machine

(SVM) algorithms to classify Keçimen and Besni raisin seeds. By utilizing a multi-metric evaluation approach—including sensitivity, specificity, and Kappa coefficient—the researchers determined that the SVM model offered a more stable and reliable performance, outperforming the Decision Tree in accurately identifying raisin cultivars. Karimi et al. [20] developed an intelligent expert system to automate the purity and quality measurement of Golden Bleached raisins using texture-based image analysis. By extracting 146 features and applying Principal Component Analysis (PCA) for dimensionality reduction, the researchers found that the SVM classifier outperformed the ANN model with an accuracy of 92.71%, demonstrating its reliability for high-precision food quality monitoring. Yılmaz et al. [21] proposed a hybrid classification framework that integrates Convolutional Neural Network (CNN) based feature extraction with traditional machine learning algorithms such as KNN, XGBoost, and SVC. Their experimental findings indicated that these hybrid models significantly outperformed standalone classical methods, demonstrating that deep learning-based representation learning enhances both accuracy and stability in raisin cultivar classification. Kılıçarslan [22] introduced a hybrid classification framework for raisin grains by integrating Rotation Forest (ROF) with Stacked Autoencoder (SAE) algorithms to enhance automated grading efficiency. By leveraging the feature learning capabilities of deep learning alongside classical machine learning ensembles, the proposed decision support system achieved an accuracy of 91.50%, demonstrating superior performance and stability compared to standalone traditional or deep learning models. Raihen and Akter [23] conducted a comprehensive comparative study of various machine learning and deep learning architectures to classify raisin cultivars based on morphological features. Their findings highlighted that gradient boosting methods, specifically LightGBM, achieved a superior accuracy rate of 98.40% with high ROC-AUC scores, suggesting that boosting-based algorithms and deep neural networks provide a highly effective framework for automated agricultural grading. Mohamed and Abdulkader [24] introduced a high-performance deep learning model designed to overcome the limitations of existing raisin classification methods. By employing a rigorous preprocessing pipeline that included normalization and Principal Component Analysis (PCA) alongside optimized regularization techniques, the researchers achieved accuracy and F-score exceeding 91%, representing a 5% performance improvement over previously reported models in the literature.

In this study, the classification performances of k-Nearest Neighbors (k-NN), Multilayer Perceptron (MLP), and Support Vector Machines (SVM) were comparatively evaluated for distinguishing between Keçimen and Besni

raisin varieties. The publicly available Raisin dataset obtained from the UCI Machine Learning Repository was used, and all experiments were conducted using WEKA software. The models were evaluated under both an 80% training–20% testing scheme and 10-fold cross-validation.

Although the dataset and machine learning algorithms employed in this study have previously been investigated in literature, the contribution of the present work lies in the systematic evaluation of k-NN, MLP, and SVM models under a unified experimental framework. Rather than proposing a new classification algorithm, the study comparatively examines algorithm-specific parameter configurations, including different neighborhood sizes for k-NN, hidden-layer neuron numbers for MLP, and kernel functions for SVM. All models were evaluated under the same experimental conditions using both an 80% training–20% testing split and 10-fold cross-validation. Their performances were assessed using accuracy, precision, recall, F1-score, Matthews correlation coefficient, confusion matrices, ROC curves, and AUC values. This consistent evaluation framework enables the relative strengths of the classifiers to be compared and provides information regarding their stability, generalization capability, and sensitivity to different validation strategies.

2. MATERIAL AND METHODS

2.1. Dataset

In this study, the Raisin Dataset was obtained from the UCI Machine Learning Repository and consists of 900 samples belonging to Keçimen and Besni raisin varieties cultivated in Turkey [25]. Each sample is represented by seven features extracted using image processing techniques:

Area: The number of pixels within the boundaries of the raisin.

MajorAxisLength: The length of the major axis.

MinorAxisLength: The length of the minor axis.

Eccentricity: The eccentricity value.

ConvexArea: The convex area.

Extent: The extent value.

Perimeter: The perimeter distance.

The dataset contains 450 samples from each class (Keçimen and Besni), thus exhibiting a balanced structure. A statistical summary of the features is presented in Table 1. The dataset was converted into the .arff format for use in the WEKA software.

Table 1. Descriptive Statistics of the Features

Feature	Mean	Standard Deviation	Min	Max
Area	87804.78	39008.42	25387	235047
Major Axis Length	430.94	116.75	225.63	941.54
Minor Axis Length	254.18	52.58	129.58	492.64
Eccentricity	0.83	0.09	0.45	0.97
Convex Area	91186.98	40797.45	26139	278217
Extent	0.70	0.05	0.45	0.84
Perimeter	1185	312.68	619	2697

2.2. Data Preprocessing

The dataset was converted into ARFF format and imported into WEKA for the classification experiments. No external normalization, standardization, feature transformation, or outlier removal filter was applied before model training. All seven original morphological features and all 900 observations were retained in order to evaluate the classifiers using the original feature representation and sample composition of the publicly available dataset. No explicit statistical outlier detection or removal procedure was applied. No observations were excluded solely because of extreme numerical feature values, since these values may represent the natural morphological variability of Keçimen and Besni raisin grains.

2.3. WEKA Software

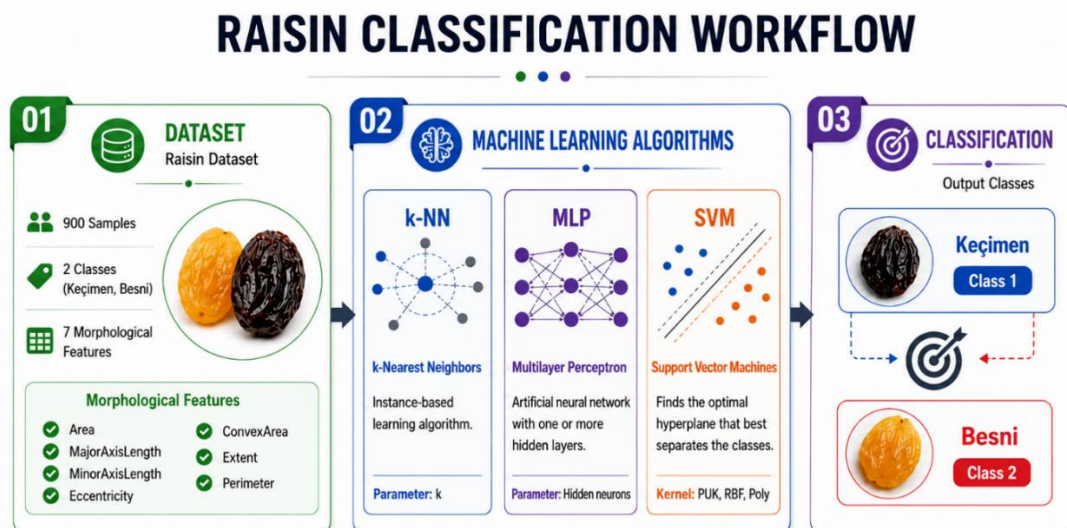
WEKA (Waikato Environment for Knowledge Analysis) is an open-source software developed for implementing machine learning algorithms and performing data analysis [26]. It provides various functionalities such as data preprocessing, classification, clustering and visualization. In this study, the dataset was processed and the algorithms were evaluated using WEKA version 3.8.5.

2.4. Machine Learning Algorithms

The k-Nearest Neighbors (k-NN) algorithm is a simple

yet effective learning method used for classification and regression problems in machine learning. It classifies a data point based on the labels of its k nearest neighbors and makes decisions using approaches such as majority voting or weighted averaging. The algorithm performs well particularly on small datasets and low-dimensional feature spaces; however, its computational cost may become high for large datasets [27]. The Multilayer Perceptron (MLP) is a machine learning algorithm based on artificial neural networks and is commonly used for both classification and regression tasks. MLP is a feedforward neural network consisting of an input layer, one or more hidden layers, and an output layer. Learning is achieved through the optimization of weights using the backpropagation algorithm [28]. The Support Vector Machine (SVM) is a powerful machine learning algorithm used for classification and regression problems. SVM aims to find the optimal hyperplane that separates data points, maximizing the margin between different classes [29]. This study aims to comparatively evaluate the effectiveness of the selected machine learning algorithms. The block diagram of the study is presented in Figure 1. The performance of the algorithms was evaluated using the 80% training–20% testing (Percentage Split) method and 10-fold cross-validation. Performance metrics including Accuracy, Precision, Recall, F1-score, and Matthews Correlation Coefficient (MCC) were used for evaluation.

Receiver Operating Characteristic (ROC) curves and Area Under the ROC Curve (AUC) values were additionally used to evaluate the discrimination capability of the classifiers. The ROC curves were generated using the prediction scores produced by the best-performing k-NN, MLP, and SVM configurations in WEKA. The Keçimen variety was defined as the positive class in all ROC analyses. AUC values closer to 1 indicate stronger discrimination between the Keçimen and Besni varieties, whereas a value of 0.5 indicates discrimination performance comparable to random classification.

**Figure 1.** Workflow of the Study

3. RESULTS

3.1. Results Obtained with the k-NN Algorithm

In this study, classification was performed using different neighbor values in the k-NN algorithm, and the obtained results are presented in Table 2. According to the 80% training–20% testing split, the classification performance of the model improved as the value of k increased. The highest classification accuracy of 90% was achieved when the number of nearest neighbors was set to $k = 7$.

As shown in the confusion matrix in Figure 2, 81 samples were correctly classified and 10 samples were misclassified in the Keçimen class. Similarly, in the Besni class, 81 samples were correctly classified, while 8 samples were incorrectly predicted. These results indicate that the k-NN algorithm provides a relatively balanced classification performance for both classes. However, a limited number of misclassifications are still observed, suggesting that there is a certain degree of overlap between the feature distributions of the Keçimen and Besni classes. Furthermore, it is observed from Table 2 that increasing the k value initially improves the model performance, while further increases may lead to slight performance degradation. This behavior can be attributed to the trade-off between overfitting and underfitting, where smaller k values tend to overfit the data, and larger k values may oversmooth the decision boundaries.

Table 2. Results of k-NN Algorithm (80% Training-20% Testing)

k	Accuracy (%)	Precision	Recall	F1-Score	MCC
1	82.78	0.828	0.828	0.828	0.656
3	86.11	0.861	0.861	0.861	0.722
5	87.78	0.878	0.878	0.878	0.756
7	90.00	0.900	0.900	0.900	0.800
9	89.44	0.895	0.894	0.894	0.789

k-NN Confusion Matrix (k=7, 80% Training - 20% Testing)

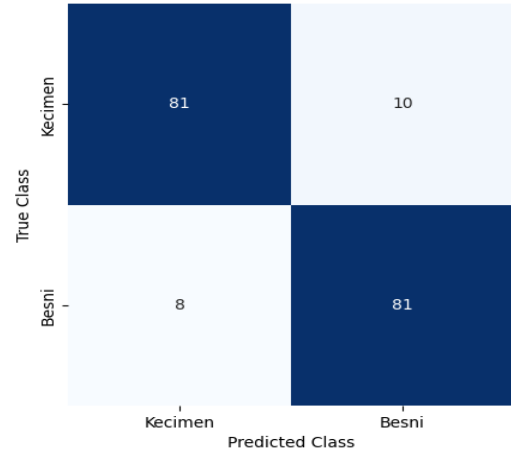


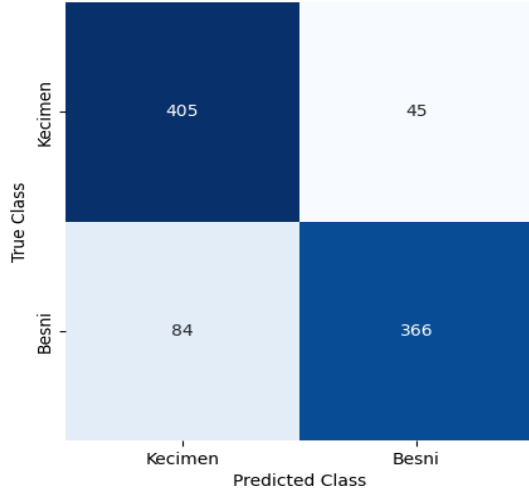
Figure 2. Confusion Matrix Obtained with k-NN Algorithm (k=7, 80% Training, 20% Test)

In this study, the performance of the k-NN algorithm was also evaluated using the 10-fold cross-validation method, and the obtained results are presented in Table 3. As can be observed, the classification performance generally improved with increasing values of k. The highest classification accuracy of 85.67% was achieved when the number of nearest neighbors was set to $k = 9$. As illustrated in the confusion matrix in Figure 3, 405 samples from the Keçimen class were correctly classified, while 45 samples were misclassified. Similarly, for the Besni class, 366 samples were correctly classified, whereas 84 samples were incorrectly predicted.

These results indicate that the k-NN algorithm provides a relatively consistent classification performance under the cross-validation scheme. However, compared to the hold-out validation approach, a slight decrease in overall accuracy is observed, which can be attributed to the more robust and unbiased evaluation mechanism of cross-validation.

Table 3. Results of k-NN Algorithm (10-fold Cross-Validation)

k	Accuracy (%)	Precision	Recall	F1-Score	MCC
1	81.44	0.814	0.814	0.814	0.629
3	82.55	0.827	0.826	0.825	0.652
5	84.22	0.844	0.842	0.842	0.686
7	85.11	0.853	0.851	0.851	0.704
9	85.67	0.859	0.857	0.856	0.716

k-NN Confusion Matrix (k=9, 10-Fold Cross-Validation)**Figure 3.** Confusion Matrix Obtained with k-NN Algorithm (k=9, 10-fold Cross-Validation)

3.2. Results Obtained with the MLP Algorithm

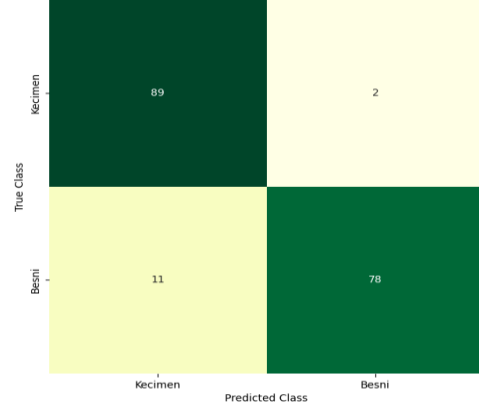
In this study, the classification performance of the Multilayer Perceptron (MLP) algorithm was evaluated using different numbers of neurons in the hidden layer. The results obtained are presented in Table 4. According to the 80% training–20% testing split, the highest classification accuracy of 92.78% was achieved when the number of neurons in the hidden layer was set to 7. As illustrated in the confusion matrix in Figure 4, 89 samples from the Keçimen class were correctly classified, while only 2 samples were misclassified. Similarly, in the Besni class, 78 samples were correctly classified, whereas 11 samples were incorrectly predicted.

These results indicate that the MLP algorithm demonstrates a high classification performance with very low misclassification rates, particularly for the Keçimen class. Compared to the k-NN algorithm, the MLP model provides improved classification accuracy and better generalization capability.

Overall, the experimental findings reveal that the MLP algorithm outperforms the k-NN approach under the hold-out validation scheme and provides a more robust and reliable classification performance on the Raisin dataset.

Table 4. MLP Algorithm Results (80% Training, 20% Testing)

number of neurons	Accuracy (%)	Precision	Recall	F1-Score	MCC
4	90.55	0.910	0.906	0.905	0.815
5	91.11	0.916	0.911	0.911	0.827
6	90.55	0.917	0.906	0.905	0.822
7	92.78	0.932	0.928	0.928	0.860
8	92.22	0.926	0.922	0.922	0.848
9	91.67	0.921	0.917	0.916	0.837
10	92.22	0.926	0.922	0.922	0.848

MLP Confusion Matrix (Number of neurons in the hidden layer: 7, 80% Training, 20% Test)**Figure 4.** Confusion Matrix Obtained with MLP Algorithm (seven neurons in the hidden layer, 80% Training, 20% Test)

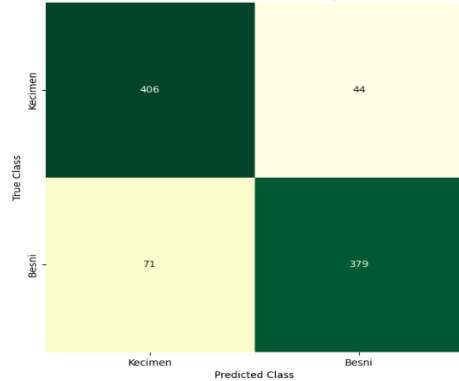
In this study, the performance of the Multilayer Perceptron (MLP) algorithm was also evaluated using the 10-fold cross-validation method. The results obtained for different numbers of neurons in the hidden layer are presented in Table 5. According to the results, the highest classification accuracy of 87.22% was achieved when the number of neurons in the hidden layer was set to 8. As illustrated in the confusion matrix in Figure 5, 406 samples from the Keçimen class were correctly classified, while 44 samples were misclassified. Similarly, in the Besni class, 379 samples were correctly classified, whereas 71 samples were incorrectly predicted.

These results indicate that the MLP algorithm provides a stable and reliable classification performance under the 10-fold cross-validation scheme. Although increasing the number of neurons generally improves the model performance up to a certain point, the slight decrease observed after 8 neurons suggests that further increasing the model complexity does not necessarily improve classification success.

Table 5. MLP Algorithm Results (10-fold Cross-Validation)

number of neurons	Accuracy (%)	Precision	Recall	F1-Score	MCC
4	86.33	0.864	0.863	0.863	0.728
5	86.44	0.865	0.864	0.864	0.730
6	86.89	0.870	0.869	0.869	0.739
7	86.89	0.871	0.869	0.869	0.740
8	87.22	0.874	0.872	0.872	0.746
9	87.00	0.872	0.870	0.870	0.742
10	87.11	0.873	0.871	0.871	0.744

MLP Confusion Matrix (Number of neurons in the hidden layer = 8, 10-Fold Cross-Validation)

**Figure 5.** Confusion Matrix Obtained with MLP Algorithm (eight neurons in the hidden layer, 10-fold Cross-Validation)

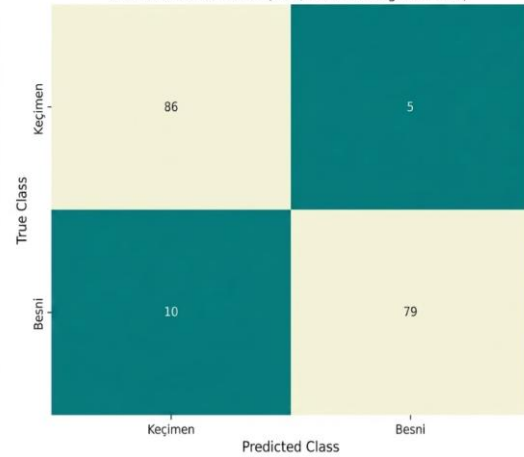
3.3. Results Obtained with the SVM Algorithm

In this study, the classification performance of the Support Vector Machine (SVM) algorithm was evaluated using different kernel functions, including Poly Kernel, Normalized Poly Kernel, PUK, and RBF. The results obtained are presented in Table 6. According to the 80% training–20% testing split, the highest classification accuracy of 91.67% was achieved using the PUK kernel. As illustrated in the confusion matrix in Figure 6, 86 samples from the Keçimen class were correctly classified, while 5 samples were misclassified. Similarly, in the Besni class, 79 samples were correctly classified, whereas 10 samples were incorrectly predicted. These results indicate that the SVM algorithm, particularly with the PUK kernel, provides a strong and balanced classification performance across both classes. Compared to other kernel functions, the PUK kernel achieves higher precision, recall, F1-score, and MCC values, demonstrating its effectiveness in capturing the underlying data distribution. Furthermore, it is observed from Table 6 that while Poly and Normalized Poly kernels provide competitive results, their performance remains slightly lower than that of the PUK kernel. On the other hand, the RBF kernel shows a significantly lower classification accuracy, indicating that it is less suitable for this dataset.

Table 6. SVM Algorithm Results (80% Training, 20% Testing)

Kernel Function	Accuracy (%)	Precision	Recall	F1-Score	MCC
Poly Kernel	89.44	0.895	0.894	0.894	0.789
Normal Poly Kernel	89.44	0.895	0.894	0.894	0.789
PUK	91.67	0.918	0.917	0.917	0.835
RBF	79.44	0.854	0.794	0.785	0.644

SVM Confusion Matrix (PUK, 80% Training-20% Test)

**Figure 6.** Confusion Matrix Obtained with SVM Algorithm (PUK, 80% Training-20% Test)

In this study, the performance of the Support Vector Machine (SVM) algorithm was further evaluated using the 10-fold cross-validation method with different kernel functions. The obtained results are presented in Table 7. According to the results, the highest classification accuracy of 86.67% was achieved using the PUK kernel. As illustrated in the confusion matrix in Figure 7, 412 samples from the Keçimen class were correctly classified, while 38 samples were misclassified. Similarly, in the Besni class, 368 samples were correctly classified, whereas 82 samples were incorrectly predicted.

These results indicate that the SVM algorithm with the PUK kernel maintains a stable and competitive classification performance under the cross-validation scheme. Although a slight decrease in accuracy is observed compared to the hold-out validation results, this is expected due to the more robust and unbiased nature of cross-validation. Furthermore, Table 7 shows that the Poly and Normalized Poly kernels produce similar performance levels, while the PUK kernel consistently achieves slightly better results across all evaluation metrics. In contrast, the RBF kernel yields the lowest performance, indicating that it is less suitable for the given dataset. Overall, the findings demonstrate that the SVM algorithm, particularly with the PUK kernel, provides reliable and consistent classification performance, achieving a good balance between model

complexity and generalization capability under the 10-fold cross-validation framework.

Table 7. SVM Algorithm Results (10-fold Cross-Validation)

Kernel Function	Accuracy (%)	Precision	Recall	F1-Score	MCC
Poly Kernel	86.44	0.866	0.864	0.864	0.731
Normal Poly Kernel	86.55	0.866	0.866	0.866	0.731
PUK	86.67	0.870	0.867	0.866	0.737
RBF	80.89	0.848	0.809	0.803	0.655

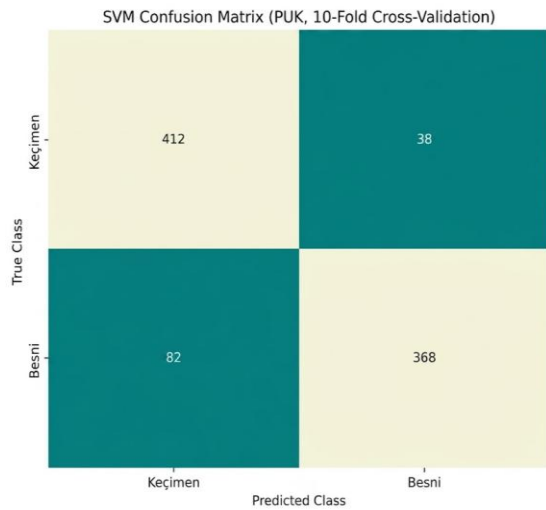


Figure 7. Confusion Matrix Obtained with SVM Algorithm (PUK, 10-fold Cross-Validation)

3.4. ROC Curve and AUC Analysis

ROC curve analysis was performed to provide a threshold-independent assessment of the discrimination capabilities of the best-performing k-NN, MLP and SVM configurations. The Keçimen variety was considered the positive class in all analyses. The AUC values obtained under the 80% training–20% testing scheme and 10-fold cross-validation are presented in Table 8.

Table 8. AUC Values of the Best-Performing Machine Learning Models

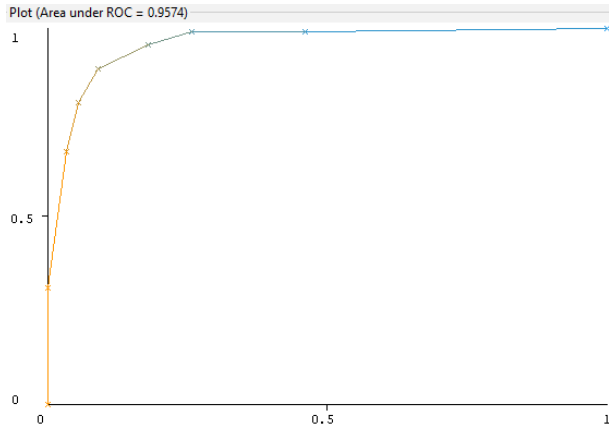
Evaluation strategy	Classifier	Best configuration	AUC
80% training–20% testing	k-NN	k = 7	0.957
80% training–20% testing	MLP	7 neurons in the hidden layer	0.968
80% training–20% testing	SVM	PUK kernel	0.962
10-fold cross-validation	k-NN	k = 9	0.912
10-fold cross-validation	MLP	8 neurons in the hidden layer	0.928
10-fold cross-validation	SVM	PUK kernel	0.914

Under the 80% training–20% testing scheme, the MLP model with seven neurons in the hidden layer achieved the highest AUC value of 0.968. The SVM model using the PUK kernel obtained a closely comparable AUC value of 0.962, while the k-NN model with k = 7 achieved an AUC of 0.957. These findings indicate that all three classifiers demonstrated strong discrimination capability between the Keçimen and Besni raisin varieties under the hold-out evaluation.

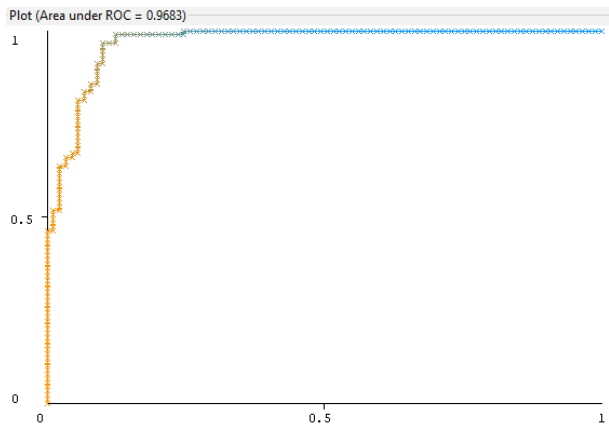
Under 10-fold cross-validation, the MLP model with eight neurons in the hidden layer again achieved the highest AUC value, reaching 0.928. The SVM model with the PUK kernel and the k-NN model with k = 9 yielded AUC values of 0.914 and 0.912, respectively. Although the AUC values were lower under cross-validation than under the fixed training–testing split, all three classifiers maintained effective discrimination performance. The decrease observed under cross-validation suggests that the fixed hold-out split may provide somewhat more optimistic performance estimates.

Figure 10 presents the classification success rates of the k-NN, MLP, and SVM algorithms obtained using the 80% training–20% testing approach. As shown in the figure, the MLP algorithm achieved the highest classification accuracy of 92.78%, outperforming both the k-NN and SVM models. The k-NN algorithm, with an optimal neighbor value of k = 7, achieved a classification accuracy of 90.00%, while the SVM algorithm using the PUK kernel reached an accuracy of 91.67%. Although all three models demonstrated high classification performance, the MLP model provided the most accurate results on the dataset. The results clearly indicate that the MLP algorithm is more effective in capturing the underlying patterns within the dataset, likely due to its ability to model complex nonlinear relationships. On the other hand, the SVM model also showed strong performance, closely following the MLP results, while the k-NN algorithm achieved slightly lower accuracy compared to the other methods. Overall, the comparative analysis demonstrates

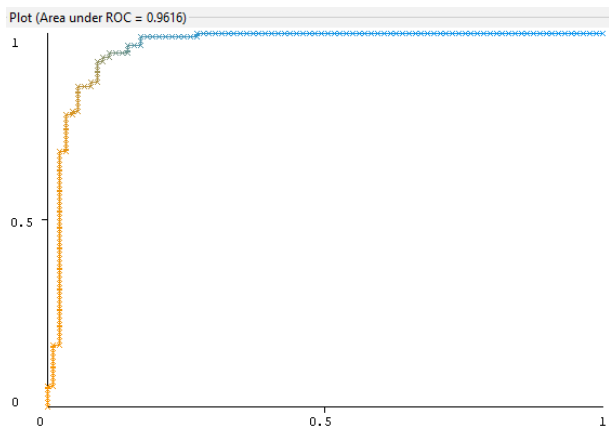
that while all models can produce reliable classification results, the MLP model offers the best performance under the hold-out validation scheme.



a.

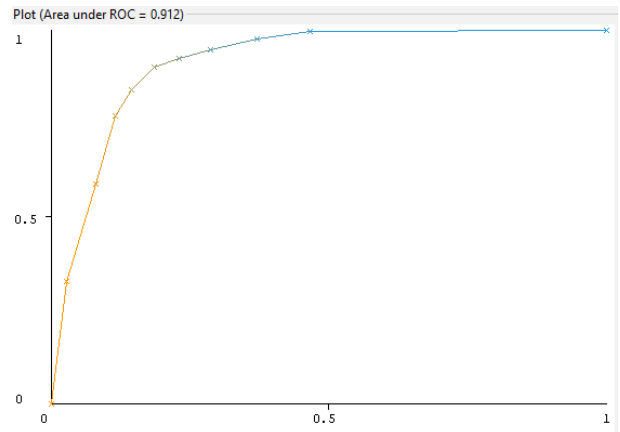


b.

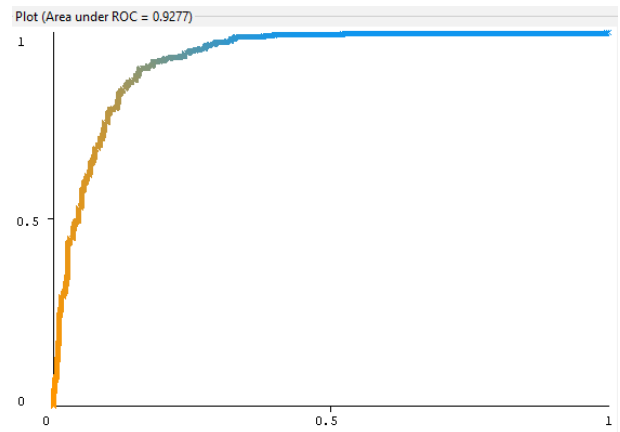


c.

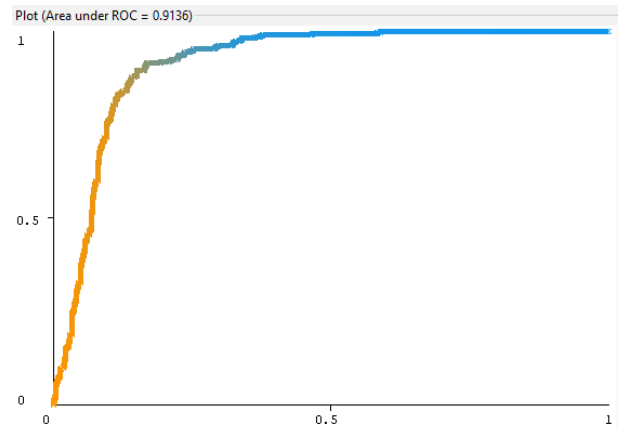
Figure 8. ROC Curves of the Best-Performing Models under the 80% Training–20% Testing Scheme: (a) k-NN, (b) MLP, (c) SVM



a.



b.



c.

Figure 9. ROC Curves of the Best-Performing Models under 10-Fold Cross-Validation: (a) k-NN, (b) MLP, (c) SVM

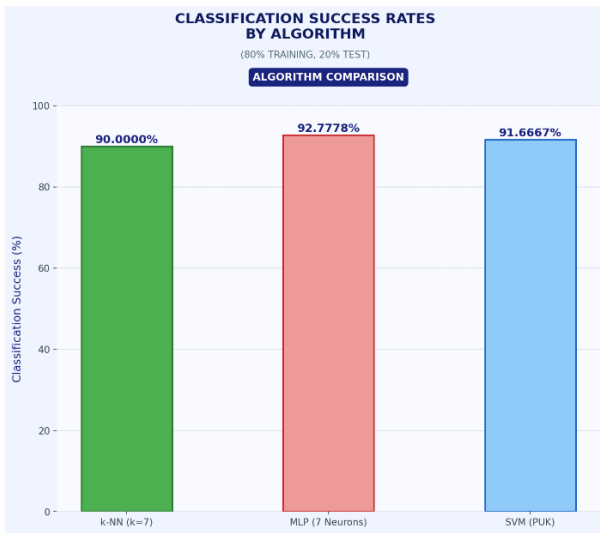


Figure 10. Classification Success Rates According to Machine Learning Algorithms (80% Training, 20% Testing)

3.5. Discussion

This study presents a comprehensive comparative analysis of three widely used machine learning algorithms—k-Nearest Neighbors (k-NN), Multilayer Perceptron (MLP), and Support Vector Machines (SVM)—for the classification of Keçimen and Besni raisin varieties based on morphological features. The experimental findings demonstrate that all evaluated models are capable of producing high classification performance; however, notable differences emerge depending on model architecture, parameter configuration, and evaluation methodology.

Accordingly, the contribution of the present study should primarily be interpreted as a controlled and consistent comparative benchmark rather than as the introduction of a novel classification algorithm. Evaluating different algorithm-specific configurations under identical experimental conditions makes it possible to examine not only the highest classification performance but also model stability, generalization capability, and sensitivity to the selected validation strategy.

Among the evaluated approaches, the MLP model achieved the highest classification accuracy of 92.78% under the 80% training–20% testing scheme when the number of hidden neurons was set to 7. This performance may be related to the ability of MLP models to represent nonlinear relationships among the input features. The observed results are consistent with existing literature, where neural network-based approaches are reported to outperform classical methods in datasets characterized by nonlinear feature interactions. Moreover, the relatively stable performance of MLP under cross-validation (87.22%) indicates a satisfactory generalization capability despite a moderate decrease compared to the hold-out evaluation.

The k-NN algorithm also demonstrated competitive performance, achieving its highest accuracy of 90.00% at

$k = 7$ under the percentage split method. This suggests that distance-based learning can effectively capture the similarity structure of the dataset, particularly due to its balanced class distribution and low-dimensional feature space. However, the decrease in performance observed under 10-fold cross-validation (85.67%) indicates that k-NN is relatively sensitive to data partitioning and parameter selection. This behavior reflects the inherent limitation of instance-based learning methods, which rely heavily on local data distributions and may be affected by variations in training subsets.

For the SVM algorithm, the PUK kernel consistently yielded the best performance among the tested kernel functions, achieving 91.67% accuracy in the hold-out validation and 86.67% under cross-validation. These findings highlight the critical importance of kernel selection in SVM-based classification problems. While polynomial-based kernels provided competitive results, the significantly lower performance of the RBF kernel suggests that it may not be well-suited for the given feature distribution. Overall, SVM demonstrated a robust and balanced classification capability, closely following the performance of the MLP model.

Another important observation in this study is the consistent decrease in classification accuracy across all models when evaluated using 10-fold cross-validation compared to the percentage split method. The lower scores obtained under 10-fold cross-validation may reflect the models' sensitivity to data partitioning and the more conservative nature of evaluation across multiple folds. Therefore, cross-validation provides a more robust estimate of generalization performance than a single fixed split; however, the observed difference should not be interpreted as definitive evidence of overfitting. When compared to recent studies in the literature, the obtained results are slightly lower than those achieved by hybrid or deep learning-based approaches. This difference can be attributed to the limited number of handcrafted morphological features used in this study. Previous research has demonstrated that integrating color, texture, or deep feature representations significantly enhances classification performance. Therefore, the incorporation of feature engineering techniques or deep learning-based feature extraction methods could further improve the effectiveness of the models.

Overall, the findings confirm that classical machine learning algorithms remain effective and reliable tools for raisin classification tasks. However, achieving optimal performance depends heavily on appropriate model selection, parameter tuning, and the use of robust evaluation strategies.

The ROC–AUC findings were consistent with the accuracy, F1-score, and MCC results. The MLP classifier achieved the highest AUC under both evaluation strategies, with values of 0.968 for the hold-out evaluation

and 0.928 for 10-fold cross-validation. The SVM model obtained the second-highest AUC values, while k-NN produced slightly lower but still satisfactory results. The decreases in AUC from the hold-out evaluation to cross-validation were 0.045 for k-NN, 0.040 for MLP, and 0.048 for SVM. These findings reinforce the observation that model performance is affected by the selected validation strategy and that cross-validation provides a more conservative estimate of generalization performance.

To position the findings of the present study within the existing literature, Table 9 provides a comparative summary of previous studies on raisin variety and quality classification. The studies are compared in terms of the dataset used, feature representation, evaluated methods, and principal reported results. Since the reviewed studies differ in their datasets, class definitions, preprocessing procedures, feature extraction methods, and evaluation protocols, the reported performance values should be interpreted cautiously rather than regarded as directly equivalent results.

Table 9. Comparison of Previous Studies on Raisin Classification

Authors	Method Used	Main Results
Çınar et al. [8]	Logistic Regression, Multilayer Perceptron, and Support Vector Machine	SVM 86.44%
Zhou [16]	XGBoost, SVM, MLP, Logistic Regression	SVM 91%
Wang et al. [10]	Hybrid image-feature extraction and Linear Discriminant Analysis	LDA – 99%
Yu et al. [11]	Color histograms, GLCM texture features, and Least Squares Support Vector Machine	LSSVM – approximately 95%
Bisri and Man [12]	Stratified sampling, shuffled sampling, and machine-learning classifiers, including Random Forest	Shuffled sampling-based models – 88.11%
Ünal et al. [14]	Decision Tree and Random Forest	Random Forest – 85.44%
Ramdhani et al. [18]	Genetic Algorithm-based feature selection and SVM	GA-optimized SVM – 87.67%
Kılıçarslan [22]	Rotation Forest and Stacked Autoencoder	Stacked Autoencoder–Rotation Forest – 91.50%
Mohamed and Abdulkader [24]	Data normalization, PCA, regularization, and a proposed deep-learning model	Proposed deep-learning model – above 91%
In this study	k-NN, MLP and SVM	MLP achieved the highest classification accuracy of 92.78%.

As shown in Table 9, the reviewed studies address raisin classification using either the publicly available Raisin

Dataset or other raisin image-, quality-, or spectroscopy-based datasets. The reported classification performances vary considerably depending on the dataset characteristics, class structure, feature representation, preprocessing procedure, model architecture, parameter optimization, data-splitting strategy, and validation protocol. Studies incorporating color, texture, spectral, or deep features generally reported higher performance than those relying solely on the original seven morphological attributes. Similarly, feature selection, ensemble learning, metaheuristic optimization, and hybrid architectures were associated with performance improvements in several studies. Therefore, the reported results should be interpreted cautiously and should not be regarded as directly equivalent because of differences in the datasets, input features, classification tasks, and experimental procedures.

In the present study, the MLP model achieved the highest hold-out accuracy and AUC values of 92.78% and 0.968, respectively. Under 10-fold cross-validation, the same algorithm remained the best-performing classifier, achieving an accuracy of 87.22% and an AUC of 0.928. Although some previous studies reported higher accuracy values, many of them employed additional color, texture, spectral, or learned features, together with feature-selection, optimization, ensemble, or hybrid modelling procedures. In contrast, the present study provides a controlled comparison of three established classifiers using the seven original morphological attributes, consistent evaluation criteria, and two different validation strategies. The difference between the hold-out and cross-validation results further demonstrates the influence of the validation protocol on the estimated classification performance.

One limitation of the present study is that no common dataset-level normalization or standardization procedure was externally applied to all classifiers before model training. Since the seven morphological attributes have substantially different numerical ranges, the absence of a unified scaling procedure may have influenced the performance of scale-sensitive algorithms. Therefore, the comparative results should be interpreted within the preprocessing and classifier settings used in this study. Future studies should systematically compare original, normalized, and standardized feature representations under the same validation protocol and should employ a separate validation set or nested cross-validation for parameter selection.

4. CONCLUSIONS

In this study, the classification performance of three fundamental machine learning algorithms—k-NN, MLP, and SVM—was systematically evaluated for distinguishing between Keçimen and Besni raisin varieties

using the Raisin dataset. The results clearly demonstrate that all three algorithms are capable of achieving high classification accuracy, confirming the effectiveness of machine learning techniques in agricultural product classification.

Among the evaluated models, the MLP algorithm achieved the best overall performance with an accuracy of 92.78% under the percentage split method, indicating its strong capability to model nonlinear relationships within the dataset. The ROC analysis further supported this result, as MLP achieved the highest AUC values of 0.968 under the hold-out evaluation and 0.928 under 10-fold cross-validation. The SVM model, particularly with the PUK kernel, also provided highly competitive results with an accuracy of 91.67%, while the k-NN algorithm achieved a maximum accuracy of 90.00% at $k = 7$. These findings indicate that although all models perform well, neural network-based approaches offer a slight advantage in terms of classification accuracy. The results obtained from 10-fold cross-validation revealed a consistent decrease in performance across all models, emphasizing the importance of robust and unbiased evaluation strategies. This highlights that relying solely on a single train-test split may lead to overly optimistic performance estimates.

The findings of this study suggest that classical machine learning algorithms can provide reliable, efficient, and interpretable solutions for raisin classification when appropriate parameter optimization and evaluation methodologies are applied. However, considering the increasing success of hybrid and deep learning-based approaches reported in the literature, there remains significant potential for further improvement.

For future research, it is recommended to explore larger and more diverse datasets, systematically compare original, normalized, and standardized feature representations, and incorporate advanced feature selection and feature engineering techniques. Deep learning-based architectures such as Convolutional Neural Networks or hybrid models may also be investigated. Additionally, integrating image-based color and texture features with morphological attributes may further improve classification accuracy and model robustness.

Declaration of Ethical Standards

The authors confirm that this study adheres to all ethical standards, including proper authorship attribution, accurate citation, appropriate data reporting and the publication of original research.

Credit Authorship Contribution Statement

The conceptualization of the research and the data collection process were carried out by Hüseyin Bulduk. The evaluation and analysis of the results were performed by Kadir Sabancı. The original draft of the manuscript was written by Hüseyin Bulduk, while the review and editing

were undertaken by Kadir Sabancı.

Declaration of Competing Interest

The authors declare that they have no competing interests.

Funding / Acknowledgements

No funding or research grants were received during the preparation of this study.

Availability of Data and Material

The dataset used in this study is publicly available from the UCI Machine Learning Repository.

References

- [1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, pp. 1097–1105, 2012.
- [2] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 4th ed. New York, NY, USA: Pearson, 2018.
- [3] M. E. Sönmez, K. Sabancı, and N. Aydın, "Classification of Wheat Rootstock and Their Hybrids According to Color Features by Machine Learning Algorithms," *International Journal of Applied Mathematics Electronics and Computers*, vol. 10, no. 2, pp. 39–48, Jun. 2022, doi: 10.18100/ijamec.1098276.
- [4] H. Bulduk and K. Sabancı, "A Hybrid Model Approach Based on Swin Transformer and EfficientNetV2 for Maize Variety Classification," *International Journal of Applied Methods in Electronics and Computers*, vol. 13, no. 3, pp. 84–92, 2025, doi: 10.58190/ijamec.2025.132.
- [5] H. Bulduk and K. Sabancı, "Performance Comparison of SVM Kernel Functions for Date Fruit Classification," *International Journal of Applied Methods in Electronics and Computers*, vol. 13, no. 3, pp. 58–64, 2025, doi: 10.58190/ijamec.2025.129.
- [6] N. Karimi, A. Arabhosseini, M. H. Kianmehr, and J. Khazaei, "Modelling of raisin berries by some physical and statistical characteristics," *International Agrophysics*, vol. 25, no. 2, pp. 141–147, 2011.
- [7] A. Semerci, T. Kızıltuğ, A. Çelik, and M. Kiracı, "Türkiye bağcılığının genel durumu," *Mustafa Kemal Üniversitesi Ziraat Fakültesi Dergisi*, vol. 20, no. 2, pp. 42–51, 2015.
- [8] İ. Çınar, M. Koklu, and Ş. Taşdemir, "Classification of raisin grains using machine vision and artificial intelligence methods," *Gazi Journal of Engineering Sciences*, vol. 6, no. 3, pp. 200–209, 2020.
- [9] M. Khojastehnazhand and H. Ramezani, "Machine vision system for classification of bulk raisins using texture features," *Journal of Food Engineering*, vol. 271, p. 109864, 2020.
- [10] S. Wang, K. Liu, X. Yu, D. Wu, and Y. He, "Application of hybrid image features for fast and non-invasive classification of raisin," *Journal of Food Engineering*, vol. 109, no. 3, pp. 531–537, 2012.
- [11] X. Yu, K. Liu, D. Wu, and Y. He, "Raisin Quality Classification Using Least Squares Support Vector Machine (LSSVM) Based on Combined Color and Texture Features," *Food and Bioprocess Technology*, vol. 5, pp. 1552–1563, 2012, doi: 10.1007/s11947-011-0531-9.
- [12] A. Bisri and M. Man, "Machine Learning Algorithms Based on Sampling Techniques for Raisin Grains Classification", JOIV: *International Journal on Informatics Visualization*, vol. 7, no. 1, pp. 7–14, 2023.
- [13] O. Şahin, "Raisin Grain Classification Using Machine Learning Models", in *Proc. 2023 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS)*, Bhopal, India, 2023, pp. 1–6, doi: 10.1109/SCEECS57921.2023.10063039.
- [14] Y. Ünal, H. Kaplan, Y. Bektaş, and M. B. Çağlar, "Classification of raisin grains variety using some machine

- learning methods”, *New Trends in Computer Sciences*, vol. 1, no. 1, pp. 62–69, 2023, doi: 10.3846/ntcs.2023.18015.
- [15] J. Guo, C. Chen, C. Chen, E. Zuo, B. Dong, X. Lv, and W. Yang, “Near-infrared spectroscopy combined with pattern recognition algorithms to quickly classify raisins”, *Scientific Reports*, vol. 12, no. 1, p. 7928, 2022.
- [16] X. Zhou, “Raisin classification based on XGBoost, SVM, MLP and logistic regression,” in *Second International Conference on Statistics, Applied Mathematics, and Computing Science (CSAMCS 2022)*, SPIE, vol. 12597, pp. 150–157, 2023. doi: 10.1117/12.2672686.
- [17] M. Dirik, “Improving Raisin Grains Classification with a Hybrid PSO-NN Approach,” in *Proc. 1st International Conference on Contemporary Academic Research (ICCAR 2023)*, pp. 17–19, 2023.
- [18] Y. Ramdhani, D. F. Apra, and D. P. Alamsyah, “Feature selection optimization based on genetic algorithm for support vector classification varieties of raisin”, *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 30, no. 1, p. 192, 2023.
- [19] W. U. Dewi, K. Nisa, and M. Usman, “Implementation of decision tree and support vector machine on raisin seed classification”, *AKSIOMA: Jurnal Program Studi Pendidikan Matematika*, vol. 12, no. 1, pp. 1309–1321, 2023.
- [20] N. Karimi, R. R. Kondrood, and T. Alizadeh, “An intelligent system for quality measurement of Golden Bleached raisins using two comparative machine learning algorithms”, *Measurement*, vol. 107, pp. 68–76, 2017.
- [21] E. K. Yılmaz, T. Oğuz, and K. Adem, “A CNN-Based Hybrid Approach to Classification of Raisin Grains,” in *Proc. International Conference on Frontiers in Academic Research*, vol. 1, pp. 460–467, 2023.
- [22] S. Kılıçarslan, “Kuru Üzüm Tanelerinin Sınıflandırılması İçin Hibrit Bir Yaklaşım,” *Mühendislik Bilimleri ve Araştırmaları Dergisi*, vol. 4, no. 1, pp. 62–71, 2022, doi: 10.46387/bjesr.1084590.
- [23] M. N. Raihen and S. Akter, “Prediction modeling using deep learning for the classification of raisin varieties-type dried fruits”, *International Journal of Mathematics and Computer in Engineering*, vol. 2, no. 1, pp. 1–12, 2023.
- [24] T. M. Mohamed and S. N. Abdulkader, “A novel deep learning model for raisin grains classification,” *Journal of Theoretical and Applied Information Technology*, vol. 101, no. 21, pp. 7052–7063, Nov. 2023.
- [25] İ. Çınar, M. Koklu, and Ş. Taşdemir, “Raisin,” *UCI Machine Learning Repository*, 2020, doi: 10.24432/C5660T.
- [26] E. Frank, M. A. Hall, and I. H. Witten, “The WEKA Workbench,” online appendix for *Data Mining: Practical Machine Learning Tools and Techniques*, 4th ed. Burlington, MA, USA: Morgan Kaufmann, 2016.
- [27] T. Cover and P. Hart, “Nearest neighbor pattern classification,” *IEEE Transactions on Information Theory*, vol. 13, pp. 21–27, 1967, doi: 10.1109/TIT.1967.1053964.
- [28] S. Haykin, *Neural Networks: A Comprehensive Foundation*, 2nd ed. Englewood Cliffs, NJ, USA: Prentice Hall, 1999.
- [29] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine Learning*, vol. 20, pp. 273–297, 1995, doi: 10.1007/BF00994018.