ISSN 2959-6157

The Investigation of Hyperparameters Influence on Fruit and Vegetable Image Recognition Performance Using SVM

Liang Chen

Department of Information and Computational Science, North China University of Technology, Beijing, China
qianchujin@ldy.edu.rs

Abstract:

This study aims to improve the efficiency of automatic classification and quality control of fruits and vegetables through image recognition technology, to achieve efficient and accurate intelligent sorting in agricultural production, reduce labor costs and improve market competitiveness. A high-quality image dataset of 36 fruit and vegetable categories from Kaggle is used in this study. The images in the dataset have been preprocessed to ensure that the data is suitable for the classification task and sets the stage for efficient training and evaluation of the model. Logistic regression was first used as the baseline model in order to compare the performance with the Support Vector Machine (SVM) model. Subsequently, hyperparameter tuning is performed to optimize the model to achieve the best cross-validation accuracy. Next, the SVM model is trained with the selected hyperparameters, and the training time is recorded. The performance of the model was evaluated in detail by confusion matrix and classification reports, and the test set was used for final validation to ensure that the model would also perform well on unseen data. The SVM model achieves an accuracy of 96% on both validation and test sets, which is a very good performance. The hyperparameters optimized by GridSearchCV (C = 10, gamma = 0.1, kernel = rbf) effectively improve the performance of the model, verifying that reasonable hyperparameter selection is crucial to the SVM model. These results show that the model has good generalization ability and potential to be applied to real-time classification tasks.

Keywords: Image recognition; logistic regression; SVM; hyperparameter tuning.

1. Introduction

Image recognition, a key area of computer vision, focuses on enabling computers to automatically understand and interpret image content. With the rapid advancement of deep learning technology, image recognition has seen significant progress and has found wide applications in fields such as healthcare, security, and agriculture. In agriculture, the sorting of fruits and vegetables has traditionally been a labor-intensive and resource-consuming task, often leading to reduced classification accuracy due to prolonged and high-intensity work. Therefore, accurate image recognition of fruits and vegetables is crucial for improving agricultural product quality control, enabling intelligent sorting, and optimizing inventory management. By quickly and accurately identifying different types and ripeness levels of fruits and vegetables, this technology can enhance agricultural production efficiency, lower labor costs, and increase market competitiveness.

Deep learning techniques have been progressing over the past few years. The yield and quality of crops can be greatly reduced due to crop diseases. For the disease of tomato crops, Rangarajan et al. used AlexNet and VGG16 models to diagnose the disease types. At the same time, the influence of the number of images and hyperparameters on the accuracy of disease classification is analyzed [1]. Due to the variety of crop diseases and pests and their fast propagation speed, in order to improve the recognition efficiency and accuracy, Wang has improved the AlexNet model, including optimizing the fully connected layer and setting different neuron nodes, which has achieved promising results [2]. In [3], in order to study the relationship between the RGB values of orange images and the sweetness of oranges, various machine learning algorithms are applied to the orange image dataset to predict the sweetness of oranges. By comparing their prediction accuracy, the logistic regression model was finally selected. In addition to crop diseases and pests, breeding has long been a problem. New crop varieties require years of testing and physical observation, collating data on their heat resistance, insect resistance and yield. In [4], an end-to-end hybrid model combining Convolutional Neural Network (CNN) and Long Short-Term Memory network (LSTM) is proposed to not only estimate the relative maturity of soybeans, but also assist plant breeding decisions. In the field of deep learning [5], Support Vector Machine (SVM) has powerful mathematical models for classification and regression. The strong mathematical foundation provides new directions for further research in the field of classification and regression. Therefore, this study aims to choose SVM model to realize the recognition of fruit and vegetable images.

In this paper, a dataset from Kaggle, created by Kritik Seth will be utilized. This balanced dataset contains a total of 36 fruits and vegetables, where each category contains 100 training images, 10 validation images and 10 test images. The excellent structure of this dataset significantly facilitated the training of the models. This study will explore the application effect of SVM algorithm in vegetable and fruit image recognition. Experiments verify the accuracy and efficiency of SVM algorithm in recognizing fruit and vegetable images. This paper provides a practical guide and reference for beginners of SVM algorithm in the field of image recognition.

2. Method

2.1 Dataset Preparation

The dataset used in this research is a well-structured collection of images from Kaggle containing 36 different categories of fruits and vegetables like apples, bananas, and potatoes to name a few [6]. These color images are not of uniform size. The images are organized into different categories with a balanced number of samples per category, and each category includes 100 training images, 10 validation images, and 10 test images, specifically designed for the image recognition task. This makes it suitable for training machine learning models aimed at classification. This dataset facilitates model performance evaluation across multiple categories, ensuring robust performance across all categories. Fig. 1 provides some sample images in the collected dataset.

Data preprocessing used in this study is performed in four steps:

Dataset Loading: The function loaded the image from the specified directory, resized it to 100×100 pixels, converted it to grayscale, and scaled the pixel values to the range [0,1]. The function also mapped each class name to an index and used this index as a label for each image.

Histogram of Oriented Gradients (HOG) Feature Extraction [7, 8]: The function extracted HOG features from an image. HOG features captured edge orientation and are particularly useful for object recognition. This function used a grid of cells of 16×16 pixels and 2×2 blocks of cells to compute the feature vector.

Label Encoding: LabelEncoder was used to convert the category labels from integers to a format suitable for machine learning models. The encoded labels were used for training, validation and testing.

Principal Component Analysis (PCA) dimensionality reduction [9, 10]: PCA was used to reduce the dimension of HOG features to 100 principal components. This step helps to reduce the computational complexity and alleviate the curse of dimensionality while preserving the fundamental patterns in the data.

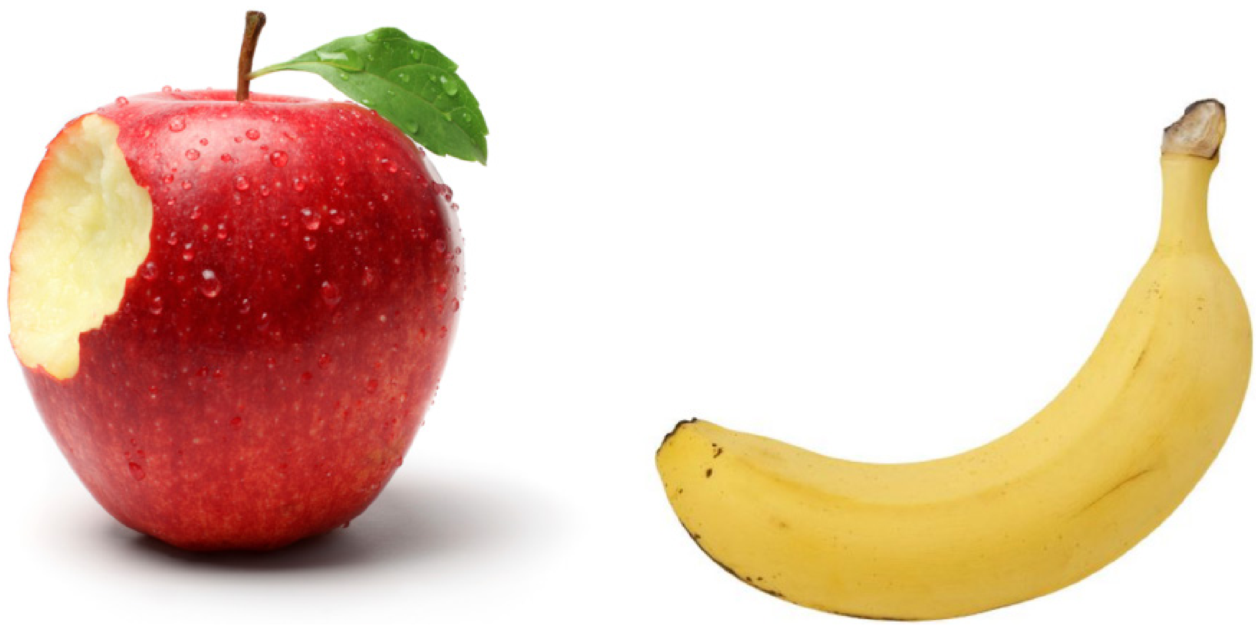

Fig. 1 Sample images in the collected dataset [6].

2.2 Machine Learning-based Prediction

2.2.1 SVM-based prediction

SVM aims to find the optimal separating hyperplane that maximizes the margin between two classes in a dataset. This margin is the distance between the hyperplane and the closest data point in each class. This margin is called the support vector. For linearly separable data, SVM finds a line (in 2D) or a plane (in higher dimensions) to separate the classes.

Given a training dataset with n samples $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$, where x_i represents the features and y_i represents the labels $y_i \in \{-1, 1\}$, SVM seeks to solve the following optimization problem:

$$\min_{w,b} \frac{1}{2} \| w \|^2 \tag{1}$$

Subject to:

$$y_i(w \cdot x_i + b) \geqslant 1 \quad \text{for all } i \tag{2}$$

w is the weight vector, and b is the bias term. The goal is to minimize $\frac{1}{2} \| w \|^2$, which maximizes the margin between the classes.

For non-linearly separable data, SVM uses kernel functions to transform the input space into a higher-dimensional space where a linear hyperplane can separate the data. Common kernels include: 1) Linear Kernel: $\langle x_i, x_j \rangle$, 2) Polynomial Kernel: $(\langle x_i, x_j \rangle + c)^d$, 3) Radial Basis Function (RBF) Kernel : $exp(-\gamma \| x_i - x_j \|^2)$, 4) Sigmoid Kernel : $tanh(\alpha \langle x_i, x_j \rangle + c)$.

In cases where the data cannot be completely separated, SVM introduces soft margins that allow some misclassification. The trade-off between maximizing the margin and minimizing the classification error is controlled by the regularization parameter C. It controls the misclassification penalty. A smaller C allows for larger intervals, while a larger C tries to classify each point correctly but can lead to overfitting.

SVM is effective in high dimensional Spaces and is efficient because it uses a subset of the training points (support vectors) in the decision function. It can be used for linear and nonlinear classification, regression, and even anomaly detection. The training time can be slow for large datasets, but it is effective for smaller or medium-sized datasets.

2.2.2 Implementation details

Baseline model setup: At the beginning, logistic regression was used as a baseline for comparison with the SVM model. This helps establish a reference point for performance.

SVM model for hyperparameter tuning: A parameter grid is defined for the SVM, which includes different values for C, gamma and kernel parameters. GridSearchCV is used to perform an exhaustive search over these hyperparameters to optimize the SVM model by selecting the combination that provides the best cross-validation accuracy.

Model training: The best SVM model is trained using the selected hyperparameters. The training time was recorded for efficiency analysis. After training, predictions are made on the validation set and the accuracy is calculated.

Model evaluation: Confusion matrices and classification reports provide detailed insight into model performance, including precision, recall, and F1-score for each class. The test set is then evaluated using the best SVM model to ensure that it generalizes well to unseen data.

Visualize and explain: Plot and box plots were used to visualize the effect of different SVM hyperparameters (C, gamma, kernel) on the validation accuracy. This helps to understand how each parameter affects the performance of the model. Confusion matrices for validation and test sets are also visualized to analyze misclassification.

Result saving and final prediction: The best SVM model and label encoder are saved for future use. Finally, some random images with their predicted and actual labels are shown to qualitatively evaluate the model predictions.

3. Results and Discussion

The validation and test sets were evaluated using the baseline model, Logistic Regression, with an accuracy of 0.5726 on the validation set and 0.5710 on the test set shown in Table 1 and Fig. 2. There was significant variance in performance between different classes. For instance, banana achieved an F1-score of 0.94, while peas only had an F1-score of 0.32. This indicates that the model was more accurate in predicting certain categories but less effective in others.

Table 1. Validation Classification Report (Baseline)

Category	Precision	Recall	F1-score	Support
Accuracy	-	-	0.57	351
Macro avg	0.61	0.57	0.57	351
Weighted avg	0.61	0.57	0.57	351

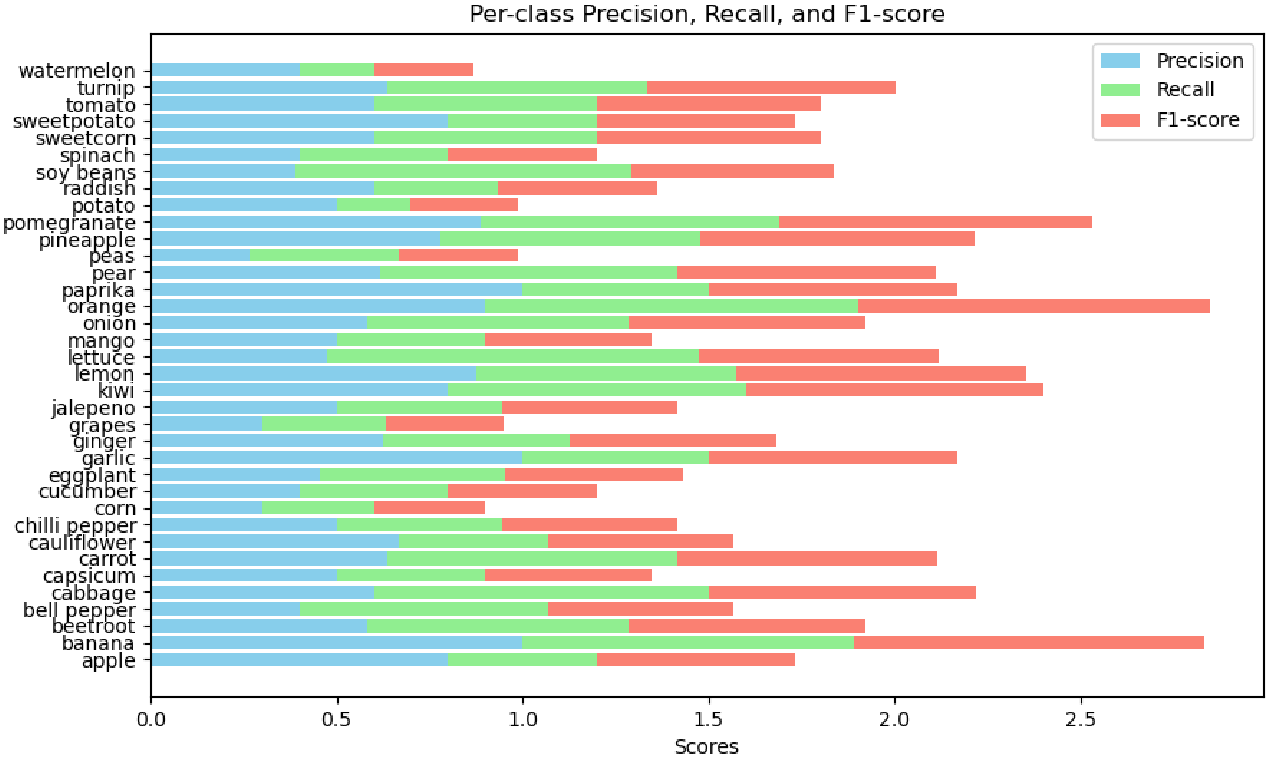

Fig. 2 Baseline model performance (Photo/Picture credit: Original)

In general, the performance of the Logistic Regression model was mediocre, but it was especially limited to complex tasks with many classes, providing a baseline for more complex SVM models.

Table 2. Validation Classification Report (SVM)

Category	Precision	Recall	F1-score	Support
Accuracy			0.96	351
Macro avg	0.96	0.96	0.96	351
Weighted avg	0.96	0.96	0.96	351

The SVM model achieved a validation accuracy of 0.96 shown in Table 2 and Fig. 3, with a prediction time of 0.15 seconds.

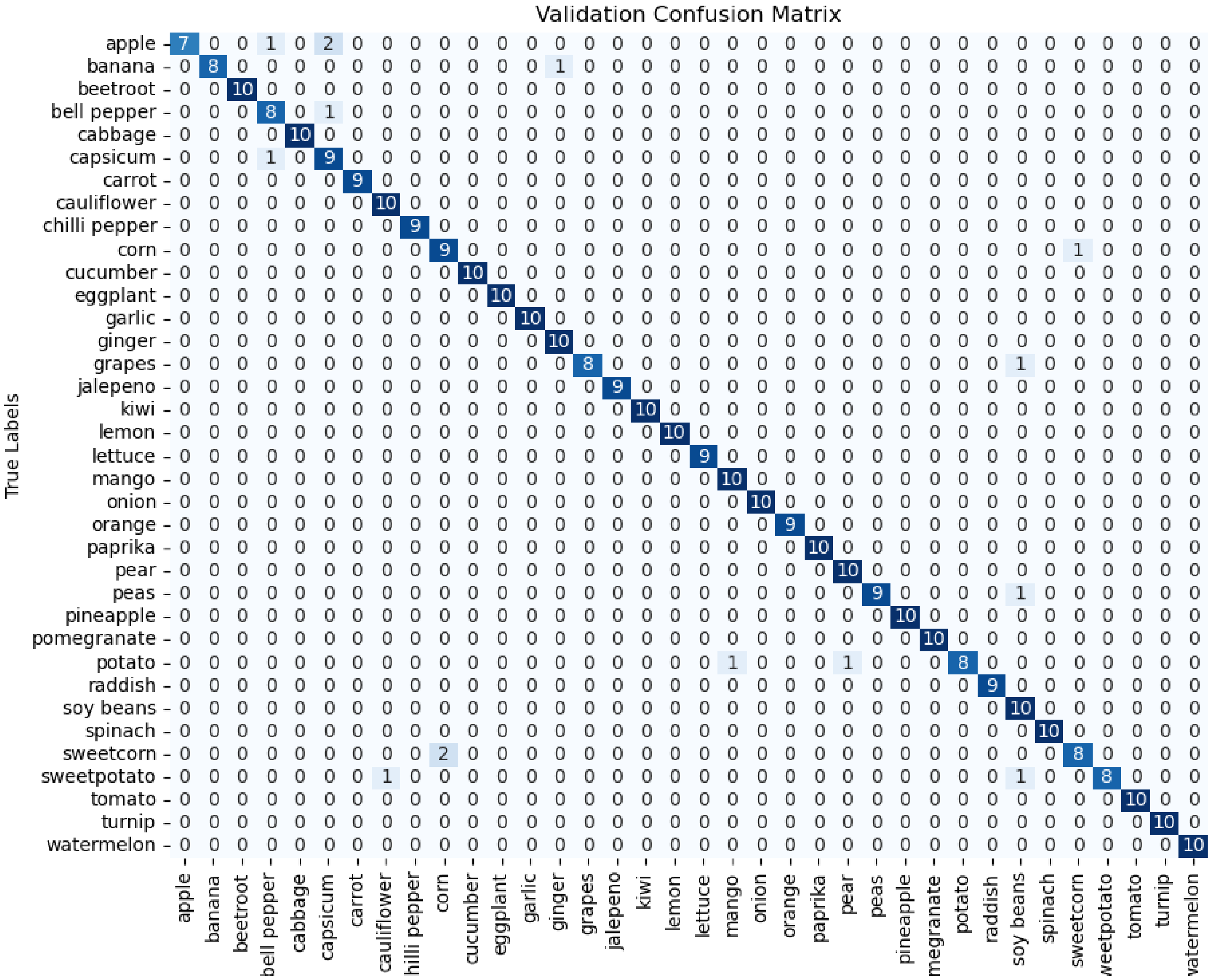

Fig. 3 Validation Confusion Matrix (Photo/Picture credit: Original)

Table 3. Test Classification Report (SVM)

Category	Precision	Recall	F1-score	Support
Accuracy	-	-	0.96	359
Macro avg	0.96	0.96	0.96	359
Weighted avg	0.96	0.96	0.96	359

The SVM model achieved a test accuracy of 0.96 shown in Table 3 and Fig. 4, with a prediction time of 0.16 seconds.

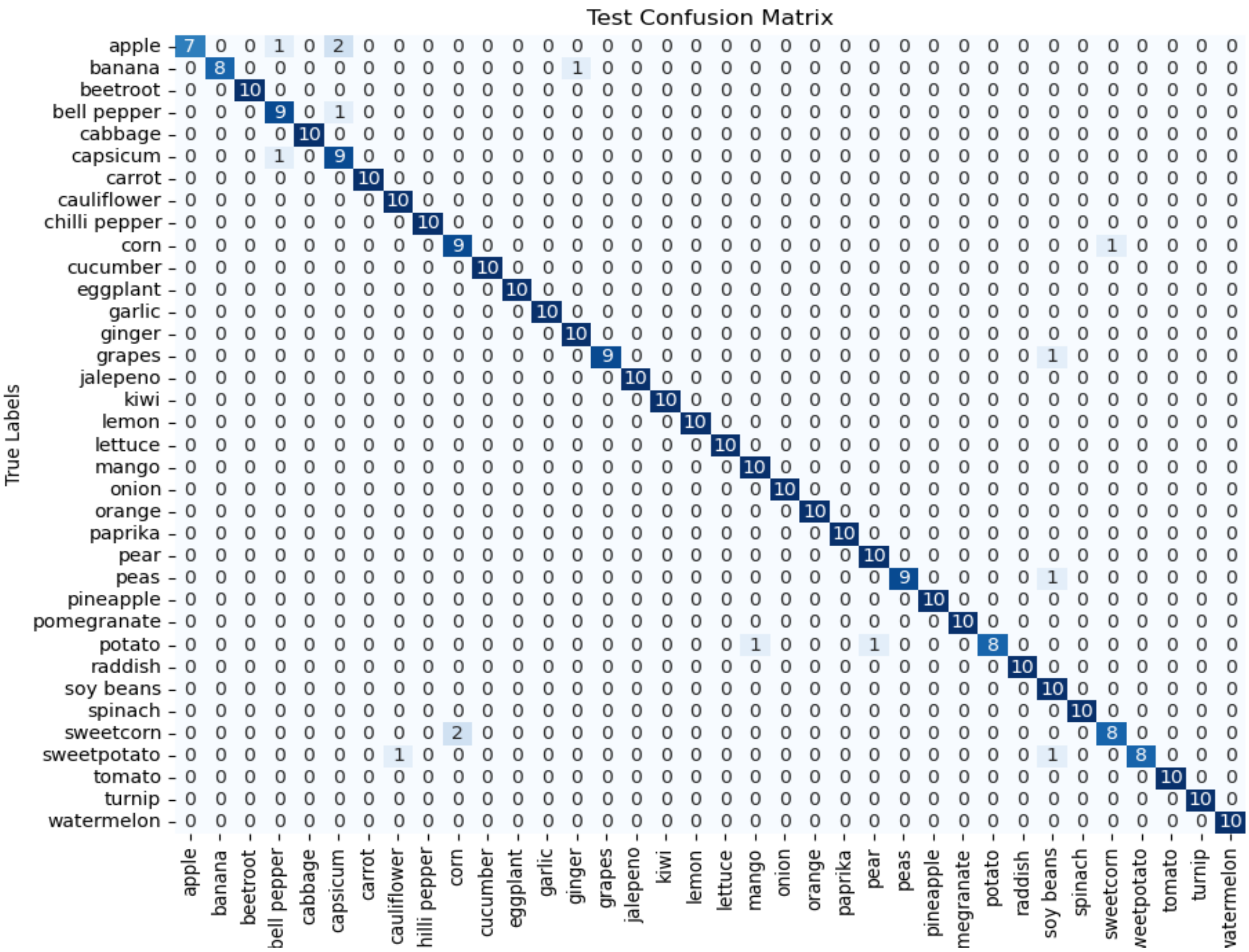

Fig. 4 Test Confusion Matrix (Photo/Picture credit: Original)

The SVM model performed very well, achieving 96% accuracy on both validation and test sets.

Precision, recall, and F1-scores were very high for most categories, although individual categories such as apple and soy beans scored slightly lower but remained within an acceptable range. The model demonstrated an accuracy of 0.96 on both the validation and test sets, which indicates consistent performance across different datasets and suggests good generalization ability. The confusion matrix revealed that the majority of predictions were correct, with only a few misclassifications, highlighting the model's strong performance across most categories. Despite class imbalance, even with small classes like bananas that had only 9 examples, the model still performed well on the relevant metrics. Additionally, the model exhibited a short prediction time, making it well-suited for application scenarios requiring real-time classification.

In the SVM hyperparameter tuning experiment, the C value, gamma value and kernel type had a significant impact on the validation accuracy of the model. Here are some summaries and analysis:

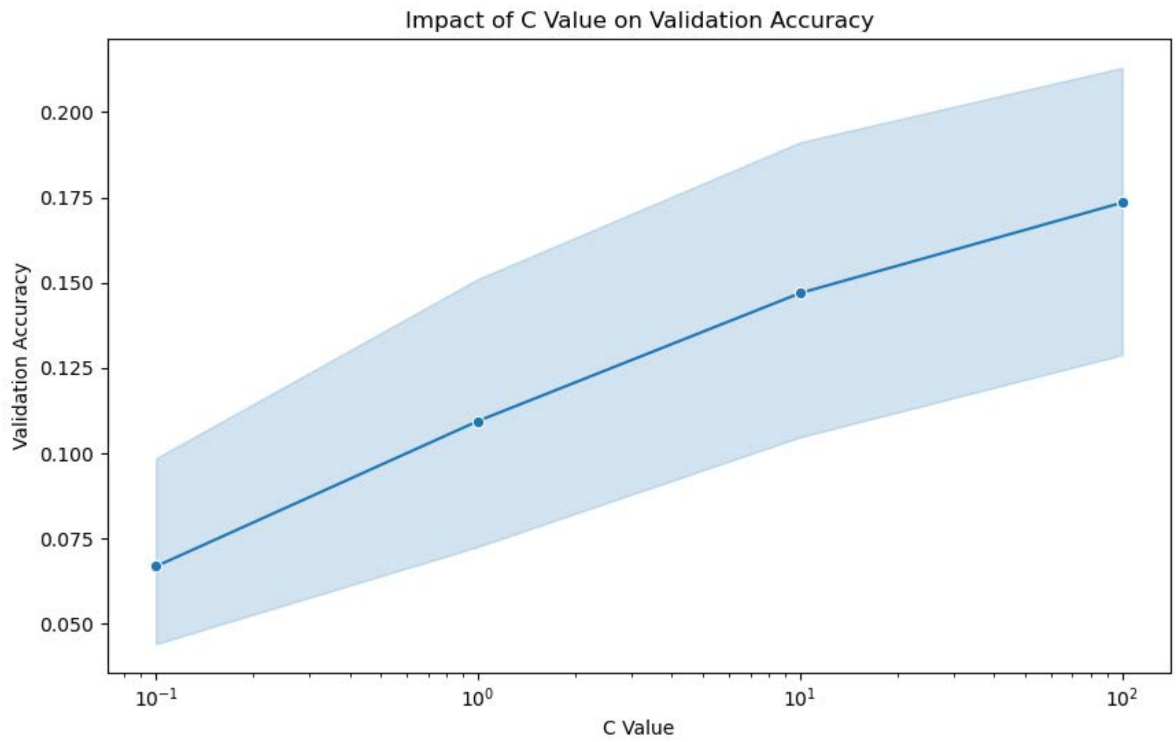

Fig. 5 Impact of C Value on Validation Accuracy (Photo/Picture credit: Original)

As can be seen in Fig. 5, the validation accuracy shows a steady upward trend with the increase of C value, indicating that a large C value helps to improve the generalization ability of the model, but a large C value may lead to overfitting.

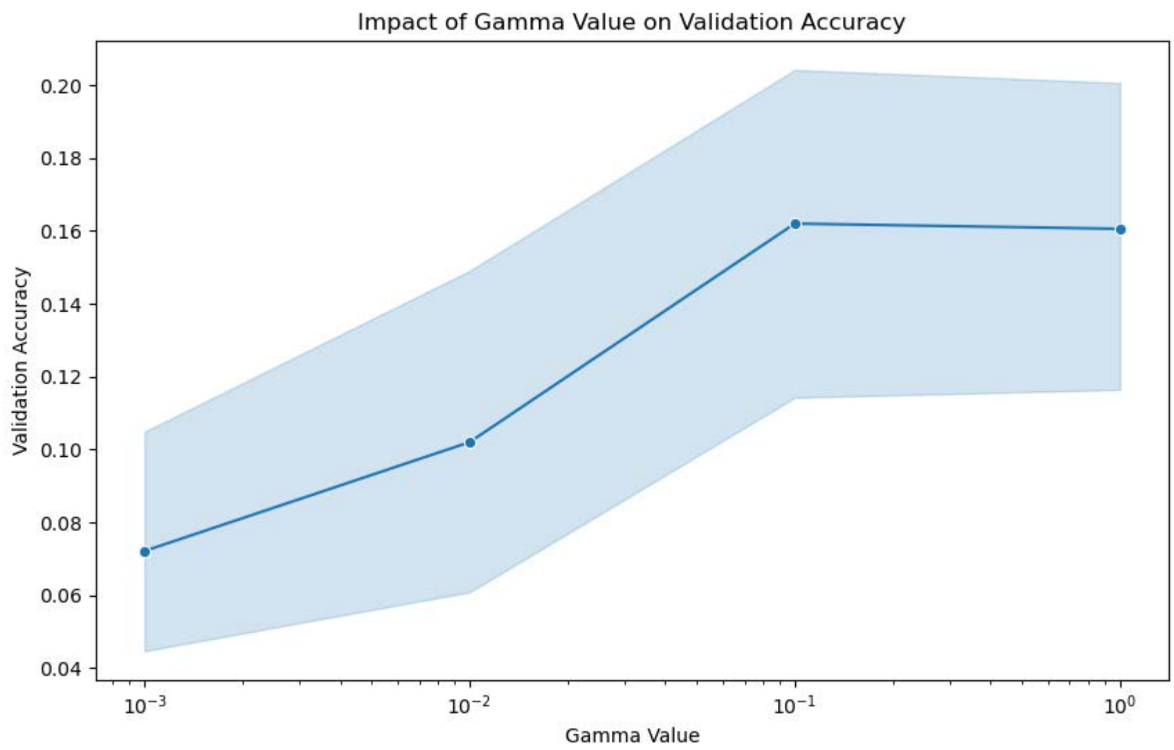

Fig. 6 Impact of Gamma Value on Validation Accuracy (Photo/Picture credit: Original)

Fig. 6 shows that the influence of gamma value on the validation accuracy is complex, the validation accuracy is the highest when the gamma value is 0.1, and too large or too small gamma value will lead to the degradation of the model performance, which reflects the trade-off between the choice of gamma value and the model complexity and accuracy.

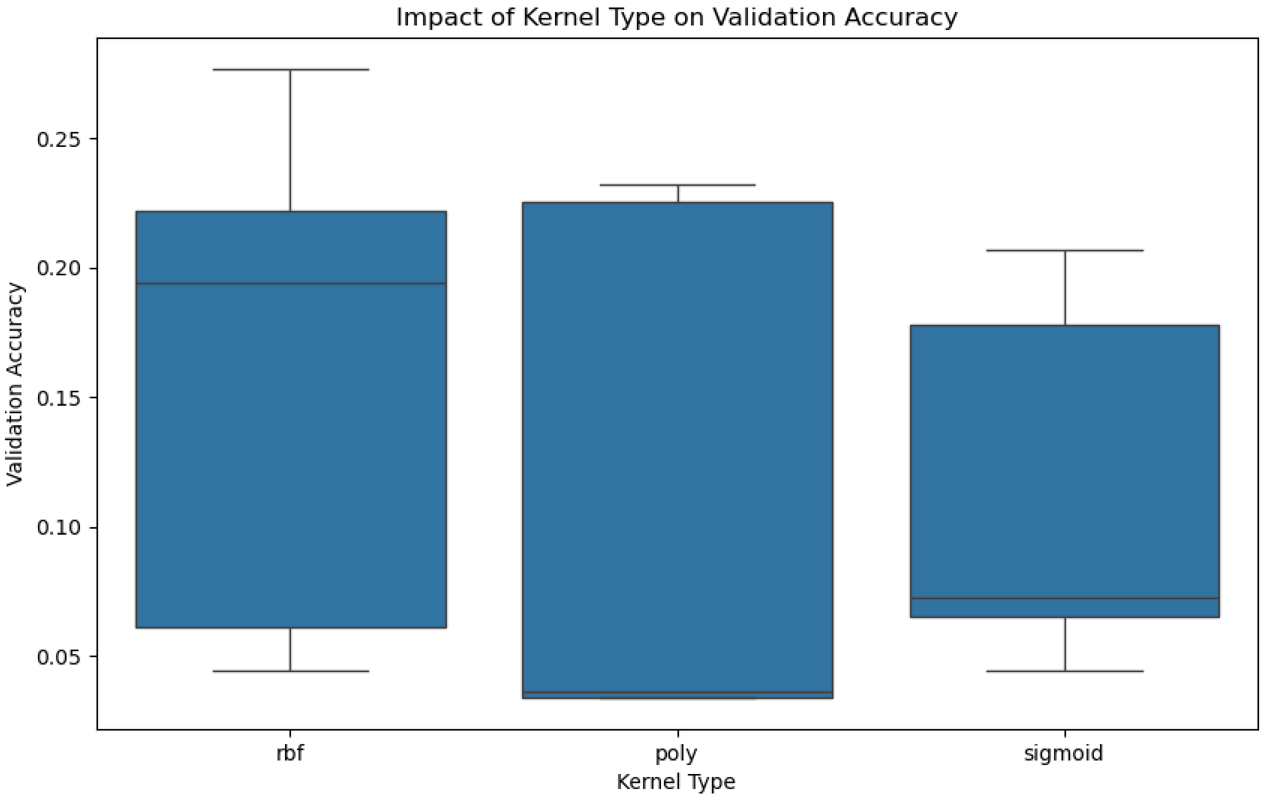

Fig. 7 Impact of Kernel Type on Validation Accuracy (Photo/Picture credit: Original)

From Fig. 7, the verification accuracy of RBF and Poly kernels is relatively high with little difference. However, the Sigmoid kernel performs significantly worse, indicating that the RBF kernel and Poly kernel are more suitable for this dataset.

Best parameter selection: Through GridSearchCV, the final best parameters are C = 10, gamma = 0.1, kernel = rbf, and this set of parameters can achieve better accuracy on the validation set. These results show that proper hyperparameter selection is crucial to improve the performance of SVM models. Through experiments, the influence of different parameters on the model is verified, and the best parameter combination is finally found.

4. Conclusion

This work used machine learning methods to recognize fruit and vegetable images. Through classification and prediction, labor costs and the consumption of human and material resources can be greatly reduced. This study used both the Logistic Regression model and the SVM model and compared the experimental data to determine the improvement of the SVM model on the classification task. After extensive experimentation, it can be found the best hyperparameters: C = 10, gamma = 0.1, kernel = rbf finally and the accuracy can reach 96%. Looking ahead, further study plans to experiment with more complex and larger datasets and use more models to find the best way to classify.

References

[1] Rangarajan AK, Purushothaman R, Ramesh A. Tomato crop disease classification using pre-trained deep learning algorithm. Procedia computer science. 2018 Jan 1;133:1040-7.

[2] Wang B. Identification of crop diseases and insect pests based on deep learning. Scientific Programming. 2022;2022(1):9179998.

[3] Al-Sammarraie MA, Gierz Ł, Przybył K, Koszela K, Szychta M, Brzykcy J, Baranowska HM. Predicting fruit's sweetness using artificial intelligence—case study: Orange. Applied Sciences. 2022 Aug 17;12(16):8233.

[4] Moeinizade S, Pham H, Han Y, Dobbels A, Hu G. An applied deep learning approach for estimating soybean relative maturity from UAV imagery to aid plant breeding decisions. Machine Learning with Applications. 2022 Mar 15;7:100233.

[5] Chandra MA, Bedi SS. Survey on SVM and their application

in image classification. International Journal of Information Technology. 2021 Oct;13(5):1-1.
[6] Kaggle Fruits and Vegetables Image Recognition, https://www.kaggle.com/code/aaronfriasr/fruits-and-vegetables-image-recognition#Fruits-and-Vegetables-Image-Recognition-%F0%9F%8D%89%F0%9F%8D%91%F0%9F%A5%A5, 2024.
[7] Dalal N, Triggs B. Histograms of oriented gradients for human detection. In 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05) 2005 Jun 20 (Vol. 1, pp. 886-893). Ieee.
[8] Déniz O, Bueno G, Salido J, De la Torre F. Face recognition using histograms of oriented gradients. Pattern recognition letters. 2011 Sep 1;32(12):1598-603.
[9] Maćkiewicz A, Ratajczak W. Principal components analysis (PCA). Computers & Geosciences. 1993 Mar 1;19(3):303-42.
[10] Daffertshofer A, Lamoth CJ, Meijer OG, Beek PJ. PCA in studying coordination and variability: a tutorial. Clinical biomechanics. 2004 May 1;19(4):415-28.