Fruit Image Classification Using Convolution Neural Networks

Yifan Yang *

School of Electrical Engineering and Computer Science, University of Ottawa, Ottawa, Canada

* Corresponding Author Email: yyang350@uottawa.ca

Abstract. Artificial intelligence has been used in many places in people's daily life, and there are more and more methods to classify objects by using deep learning. However, at present, the technique of fruit classification is mainly manual classification, and the accuracy of mechanical classification needs to be improved. The ideas for fruit recognition are primarily focused on distinguishing the shape, texture, and colour of fruits. Therefore, this paper uses deep learning technology since the performance of deep learning in the field of computer vision is better than that of traditional machine learning. The convolutional neural network (CNN) of deep learning will automatically learn the features of different fruit images to establish a model for predicting fruit types. In this paper, four CNNs models with different structures are compared with the prediction results for 131 different types of fruit. The data showed that the best model provided 98.2% accuracy.

Keywords: deep learning, fruit classification, convolutional neural network, image classification.

1. Introduction

The concept of a healthy diet has become popular in many places of the world, leading to a rapid increase in the demand for fruit [1]. Therefore, various regions need more and different types of fruits to meet the needs of customers. The increased variety of fruit can lead to increased costs for grocery stores or fruit distributors to sort and identify fruits. Store sellers' traditional solution is labelling different fruits with their own species-specific bar codes. However, this is a time-consuming task, which also means that fruit sorting is done manually. This makes labor costs increase, and efficiency cannot be guaranteed. The use of bar codes also increases the cost of materials.

Since most fruits can intuitively reflect their own characteristics through RGB images, computer vision can be widely used in fruit classification. Many computer vision technologies can obtain and process large amounts of digital information through images. Traditional machine learning methods can provide good feature extraction, such as support vector Machine (SVM), Random Forest and K-th Nearest neighbor (KNN) [2]. The extraction of features is one of the two crucial components of a fruit recognition model. The traditional machine learning method has a good prediction accuracy when processing images with fewer features. SVM is a powerful classification algorithm in machine learning, which is usually used in different classification scenarios. In the study of Chaganti and Nanda et al. [3], when given a very small dataset, the use of SVM can provide a good accuracy of 93%. However, once the dataset is enlarged, the accuracy of using SVM decrease to 82%. In their study, they also mentioned that using the CNN method, the accuracy of model prediction is directly proportional to the length of the dataset. CNN's accuracy for the biggest dataset is 93.57%, which is significantly greater than SVM's accuracy of 82%. In another study by Baranwal and Jaiswal et al. [4] on the classification of brain tumor images, the prediction accuracy of CNN in various types of tumors was higher than that of SVM. When using small sample data sets, traditional machine learning algorithms can provide fast and high-accuracy prediction models, but when using large sample data sets, CNN can provide better prediction models also with better tolerance and robustness [5].

Through pre-training models, deep learning automatically extracts and continuously optimizes features, which can quickly process a large amount of data and achieve better performance and higher accuracy. In this paper, a CNN model is established based on computer vision and deep learning to classify and train tens of thousands of fruit images.

2. Deep Learning

Deep learning is a subarea of machine learning, which is one type of algorithm. It represents and learns data based on the neural network architecture that mimics the human brain. Deep learning is a multi-layer algorithm that uses many nonlinear processing units (neurons), and each layer processes the input data by self-learnable parameters. Deep learning has demonstrated higher accuracy and performance in comparison to conventional classification methods like Support Vector Machine and Kth Nearest Neighbor [6]. It can be implemented in the field of machine learning to model patterns such as sound, image, and line, also it is a statistical probability model. In deep learning, modelling is a self-learning process, and after this step, the model can make predictions about new data. Convolutional Neural Network (CNN) has shown to be the most effective deep learning model in the area of image analysis.

The major benefit of the image recognition method based on deep learning over conventional image classification methods is that it can automatically learn more abstract data features through deep architecture, without the need to design detailed synthetic features for specific picture data or classification models, thereby significantly enhancing the classification effect. Deep learning can be separated into the generative model, the discriminative model, and the hybrid model, according to the learning model [6].

2.1. Convolutional Neural Network

A convolutional neural network (CNN) is a type of feed - forward neural network. CNNs are one of the most prominent algorithms for deep learning [7]. Convolution by feature mapping neural network framework provides two groups of feature extraction. Feature mapping is the purpose of dimension reduction, is the feature vector map to high dimension space to low dimension or one-dimensional characteristics of the data process. Each of the convolution CNN mapping layers contains multiple features, and also is to have multiple planes, since each mapping neuron is on the surface of the weights of sharing. Therefore, the parameters of a convolutional neural network are greatly reduced. Feature extraction is a process of interconnecting the input of each node of the convolution layer with multiple nodes of the previous convolution layer, and synthesizing the information taken from the convolution layer to obtain the global information at the highest level. The CNN realizes the picture feature extraction through the convolution kernel and pooling layer.

2.2. CNN Layers

Convolutional neural network structure generally contains multiple network layers, which are composed of the input layer, convolution layer, activation function layer, pooling layer and fully connected layer. The function of the input layer of a convolutional neural network is to input the training data into the CNN. Generally, the input is an image, which can be represented by a matrix and is the direct data of CNN for convolutional operation [8]. The convolution layer's objective is to extract features from the input neurons. It collects picture features using the weighted average of the convolution kernel and then generates feature images via the convolution process. The purpose of using the activation function is to solve the problem of inadequate Linear model expression power by utilizing nonlinear factors and approximating any nonlinear function through a convolutional neural network. Common activation functions include Reasonable Satisfaction Linear Unit (ReLU), Tanh, Sigmoid, etc. The main reason for using the pooling layer is to screen information and select features from the previously generated feature images, to decrease the training parameters of the network. The fully connected layer's function is to aggregate the retrieved feature images from the pooling layer by connecting nodes between adjacent layers, and then select different loss functions as the objective functions of the convolutional neural network for classification, regression or ranking [8].

3. Tools

This study uses Keras for the purpose of realizing the CNN model establishment of fruit classification. The top five Kaggle-winning teams utilized the Keras framework for deep learning the most. Keras is a vital component of the highly integrated TensorFlow 2 ecosystem, providing support for all phases of the learning workflow, including data administration, hyperparameter training, and deployment options [9]. Keras is straightforward but not simplistic since it lessens the cognitive load on developers, allowing them to concentrate on specific aspects of the issue.

4. Dataset

The dataset is essential to the success of the experiment since image classification relies on the information characteristics in the image to distinguish and categorize various items. In the study of visual object recognition, including picture categorization, recognition, processing, and behaviour analysis, image datasets are crucial. The fundamental building blocks of neural network training are general and niche data sets. For researchers, data sets are crucial resources. Results can only be more accurate by training and testing with high-quality and clear data sets.

As mentioned above, this study needed a dataset that not only provided multiple fruit images, but also a large number of images for each type of fruit to train on. In real life, the fruit pictures we can extract will not be displayed in a fixed posture. When conducting deep learning, we need to include every part of the fruit in the training set. The Kaggle Fruit 360 dataset meets all the attributes which are shown in Table.1 [10]:

Table 1. Dataset Attributes

Attributes	Content
Training set size	67,692 images, one fruit per image
Testing set size	22,688 images, one fruit per image
The number of classes	131 fruits
Image Size	100 x 100 pixels
Image Color	RGB

An example of the training set of Kaggle Fruit 360 is shown in Figure 1.

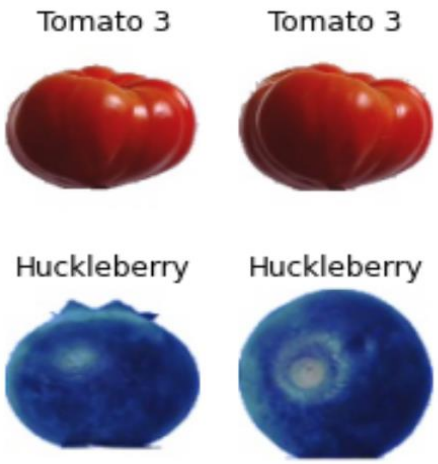


Figure 1. Example images of the training set of Kaggle Fruit 360

4.1. Data pre-processing

Deep convolutional neural networks have demonstrated exceptional performance on a range of computer vision applications. However, to prevent overfitting, these networks significantly rely on massive data.

As mentioned above, pictures of fruits may appear in various forms in real life, so image preprocessing can help the trained model to have better tolerance, add variety to the training set, and create new training examples from the existing dataset [11].

For the Fruit 360 training dataset, the following image augmentation techniques will be randomly applied to each fruit image.

4.1.1 Rescaling

Rescaling is the function of each pixel value of the images that are multiplied by the scale factor, the operation in all other transform performed before operation. The original image contains RGB coefficients ranging from 0 to 255, but such numbers are too big for the CNN model with a fixed learning rate. The object numbers are rescaled within the range of 0 to 1.

4.1.2 Shearing

As seen in Figure 2, shearing is a miscutting transformation. An image is sheared by displacing X-coordinate (or Y-coordinate) correspondingly, but pixels of another coordinate are not changed. The shift's magnitude is related to the vertical distance between the point and the X-axis (or Y-axis).

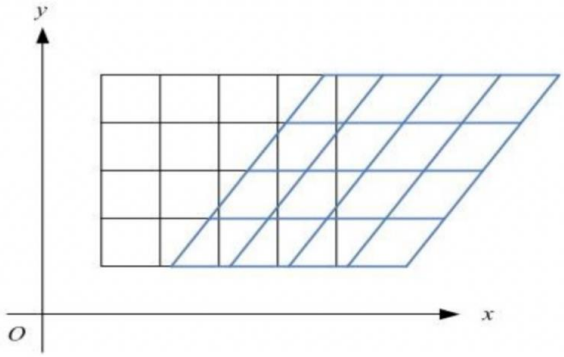


Figure 2. Example of Shearing Transformation

4.1.3 Flipping

Flipping is meaning that an image is rotated on a horizontal or vertical axis.

4.1.4 Zooming

Zooming allows the image to be enlarged in a long or wide direction. It can be understood as resizing in a certain direction, so this parameter can be a number or a list. When a number is given, the image is enlarged and shrunk to the same extent in both directions at the same time. When a list is given, it stands for [width_zoom_range, height_zoom_range]. The length and width are expanded and shrunk to different degrees respectively when the parameter is larger than 0 but less than 1, the zoom-in operation is carried out; when it is more than 1, the zoom-out operation is carried out.

5. Model Building

In this study, when constructing CNN, a variety of different structures of CNN and parameters are compared to try to achieve the best accuracy, including the learning rate, the number of convolutional layers and the number of training rounds (epochs).

5.1. Input Layer

Using the preprocessed 100 X 100 pixel RGB 3 channel matrix of the fruit image is as input x.

5.2. Convolutional Layer

As shown in Figure 3, in the convolutional layer, the filter convolves the data of the fruit image to obtain the feature map. Since there are many fruits with similar shapes, colours and textures, it is very

difficult to distinguish them, so a 3x3 convolution kernel is chosen in this study. Then the convolution kernel size influences the extraction of features from picture features. Some retrieved feature pictures may be invalid if the selected convolution kernel is of insufficient size. On the contrary, if it is too large, the complexity of the extracted features from the input image or feature map will exceed the computational ability of the convolution kernel. The large convolution kernel size has another disadvantage: overfitting [12]. Overfitting usually happens is when a machine learning algorithm learns so much from the provided training data. It loses its ability to generalize.

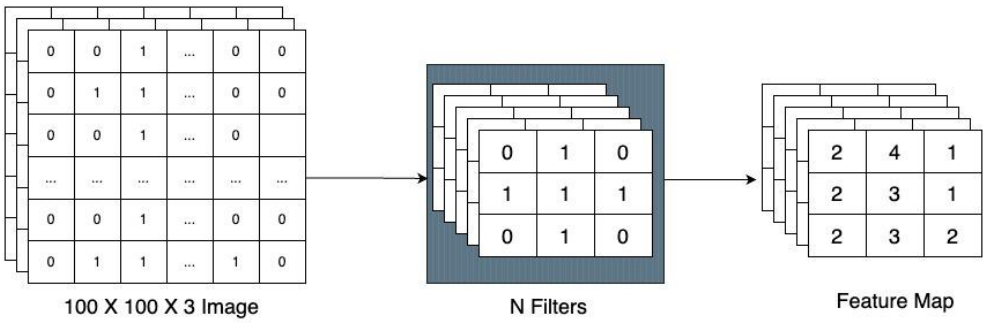


Figure 3. An image example of the structure of the convolutional layer of this study

Firstly, the 100X100 pixel map of the input layer is convolved in this study, and N filters with 3X3 convolution kernels are used to process the input. Finally, N feature maps are obtained and transferred to a next convolutional or pooling layer.

Secondly, this study will use multiple convolution layers. In addition to the first layer convolution which is the handling of the input layer, other layers of convolution are the output of the last layer feature maps to generate the corresponding thumbnail. Zooming in and out on the input image characteristics reduces the image resolution, decreases the number of parameters, and decreases the size of a feature vector.

Each convolutional layer is computed through the ReLU activation function. The mathematical formulation of this layer is expressed as in Equation 1.

$$z = f(W \bullet x + b) \tag{1}$$

In Equation 1, x is input, b is bias, f(x) is ReLU activation function, W is convolution kernel, and z is output.

5.3. Pooling Layer

Pooling is to select some way to decrease the dimension of the input matrix and compress it to speed up the computation. This study uses Max Pooling with 2 x 2 filters and stride 2. The example pooling process is shown in Figure 4. Max Pooling means that several feature values are extracted from a Filter, and only the largest Pooling layer is obtained as the retained value, while all other feature values are discarded. The greatest value signifies that just the strongest feature among these features is maintained, while the others are removed. The objective is to limit the size of model parameters and eliminate the problem of overfitting [13].

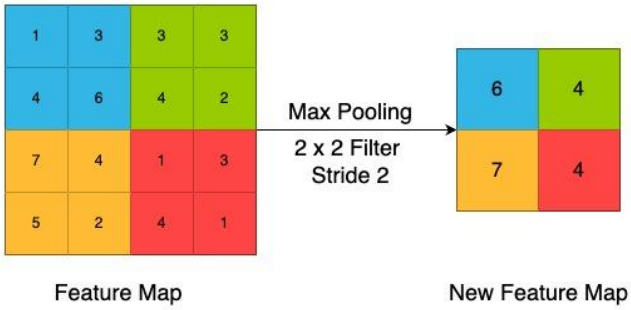


Figure 4. An image example of Max Pooling with 2 x 2 filter and stride 2

5.4. Flatten Layer

Since the Fully Connected Layer needs one-dimensional input, this study is using Flatten Layer to reshape the output of the Convolution Layers. Flattening is the transformation of a multidimensional matrix into a one-dimensional matrix. The flattening process is shown in Figure 5.

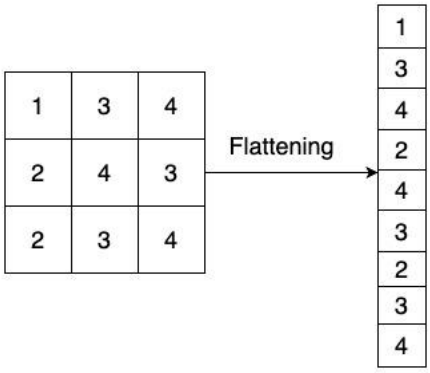


Figure 5. An image example of Flattening

5.5. Fully Connected Layer and Output Layer

A fully connected layer (FCL) shows that each neuron will connect to all neurons in the layer after being combined with the features extracted from the previous layer. The ReLU activation function is used for fully connected layers that are not the final layer.

The output layer is a type of FCL, which uses the same quantity of nodes as the fruit types as the classification output and uses the Softmax activation function. The input values are normalized to a vector of values that follow a probability distribution that sums to one in SoftMax regression. The SoftMax expression is shown as equation 2. For example, based on a set of logits [3.0, 2.0, 1.0] with the Softmax activation function, we can get an array of probabilities [0.67, 0.24, 0.09]. Therefore, at the selecting final node, the model selects the node with the first high probability (the one with the largest value) as the prediction target.

$$S(y_i) = \frac{e^{y_i}}{\sum_i e^{y_j}} \tag{2}$$

In Equation 2, y_i is logits score input.

5.6. Dropout

Dropout means that during the training process of a CNN, a neural network unit is temporarily removed from the network based on a predetermined probability. When a sophisticated neural network is trained on a short data set, it is susceptible to overfitting. To prevent overfitting, the neural network's performance can be enhanced by keeping feature detectors from interacting (Dropout) [14].

5.7. Learning Rate

In supervised learning and deep learning, the learning rate is a crucial parameter because it affects whether and when the objective function converges to the global minimum. A reasonable rate of learning permits the goal function to reach a global minimum within a sufficient time period.

A too small learning rate will make the converging process moves slowly. It makes the training curve does not descend. A too lager learning rate will make the training process would overshoot the minimum and have oscillations. It makes the training curve does not converge [15].

In this study, the method of dynamically reducing the learning rate is chosen to find the optimal learning rate to jump out of the local minimum.

When validation loss was not significantly reduced in three epochs, it will activate to reduce the learning rate.

At the beginning of the process, the learning rate is 0.01, each reduction is 50%, and the minimum learning rate is 0.0001.

5.8. Loss Function

Supervised learning in machine learning is to learn the relation between input x and output y, when given a series of training samples (x, y). When giving a X is known, the model will give a predicted Y. A loss function is applied to calculate the variance of the model's output and X's actual value [16].

This study uses Cross Entropy as the loss function. In multivariate classification problems, Cross Entropy is often used. the cross-Entropy expression is shown as Equation 3.

$$H(p,q) = -\sum_x p(x) \log q(x) \tag{3}$$

In Equation 3, p(x) is real distribution, and q(x) is predicted value.

6. Results

The test environment used in this study is shown in Table.2.

Table 2. Test Environment

Component	Name
CPU	Apple Silicon M1 Pro 10-core CPU
GPU	Apple Silicon M1 Pro 16-core GPU
Memory	16GB
System	macOS Monterey 12.5

In this paper, when constructing CNN, a variety of different structures of CNN and parameters are compared to try to get the highest accuracy. For example, the learning rate, epoch and convolutional layer structure can be involved.

Table.3 shows the training results of the CNN structures tried in this study. In this table, CL represents Convolution Layer, and FCL represents Fully Connected Layer.

Table 3. CNN Structures with Training time and Accuracy

CNN Structure	Test Rounds	Epochs	Average Time Consumption	Average Training Accuracy	Average Test Accuracy
Input Layer + CL1(filter = 32, kernel size = 3 X 3, activation= ReLU) + MaxPooling1(filter = 2 x 2, stride= 2) + CL2(filter = 64, kernel size = 3 X 3, activation= ReLU) + MaxPooling2(filter = 2 x 2, stride = 2) + CL3(filter = 128, kernel size = 3 X 3, activation= ReLU) + MaxPooling3(filter = 2 x 2, stride = 2) + Flatten Layer+	3	35	47 mins	87.8%	83.4%

FCL (units = 1024, activation=ReLU)+ Dropout (0.3) + Output Layer (units = 131(number of fruit types), activation=SoftMax)					
Input Layer + CL1(filter = 16, kernel size = 3 X 3, activation= ReLU) + MaxPooling1(filter = 2 x 2, stride = 2) + CL2(filter = 32, kernel size = 3 X 3, activation= ReLU) + MaxPooling2(filter = 2 x 2, stride = 2)+ CL3(filter = 64, kernel size = 3 X 3, activation= ReLU) + MaxPooling3(filter = 2 x 2, stride = 2) + Flatten Layer+ FCL (units = 1024, activation=ReLU) + Dropout (0.3) + Output Layer (units = 131(number of fruit types), activation= SoftMax)	3	35	51 mins	83.6%	79.2%
Input Layer + CL1(filter = 16, kernel size = 3 X 3, activation= ReLU) + MaxPooling1(filter = 2 x 2, stride = 2) + CL2(filter = 32, kernel size = 3 X 3, activation= ReLU) + MaxPooling2(filter = 2 x 2, stride = 2) + CL3(filter = 64, kernel size = 3 X 3, activation= ReLU) + MaxPooling3(filter = 2 x 2, stride = 2) + Flatten Layer+ FCL1 (units = 1024, activation= ReLU) + Dropout (0.3) + FCL2 (units = 256, activation= ReLU) + Output Layer (units = 131(number of fruit types), activation= SoftMax)	3	35	52 mins	91.4%	87.9%

Input Layer + CL1(filter = 32, kernel size = 3 X 3, activation= ReLU) + MaxPooling1(filter = 2 x 2, stride = 2) + CL2(filter = 64, kernel size = 3 X 3, activation= ReLU) + MaxPooling2(filter = 2 x 2, stride = 2) + CL3(filter = 128, kernel size = 3 X 3, activation= ReLU) + MaxPooling3(filter = 2 x 2, stride = 2) + Flatten Layer+ FCL1 (units = 1024, activation= ReLU) + Dropout (0.3) + FCL2 (units = 256, activation= ReLU) + Output Layer (units = 131(number of fruit types), activation= SoftMax)	3	35	58 mins	98.2%	94.6%

The accuracy of the results of each model was further verified by conducting three rounds of tests on CNNs with different structures and obtaining the average value of all results. To find the best CNN model for this study by comparing the training accuracy and testing accuracy. Time is a second consideration when encountering similar accuracy.

Finally, according to the experimental data, the convolutional layer depth of CNN which is set as 3 layers (each layer is 32,64 and 128) and the two fully connected layers with dense 1024 and 256, the accuracy of training is optimal. The CNN structure is shown in Figure 6.

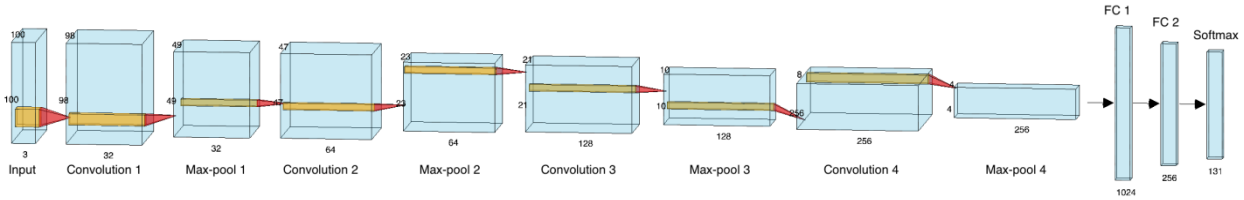


Figure 6. Structure of the Optimal CNN model of this Study

7. Conclusion

In this paper, a convolutional neural network based technique for classifying fruit images is provided. CNNs with different structures were trained according to the dataset of Kaggle 360 Fruit, and the optimal model was found by comparing the final training and testing accuracy. The experimental results show that the optimal model can achieve 98.2% accuracy for fruit image classification. In a large number of training data sets, this result outperforms the traditional machine learning methods. Due to the computational force constraint of the equipment, the epoch of this experiment was set to 35, and it was one of the future improvements of this study that more rounds of training could not be conducted. Through this study, it is also found that the training results of CNN are highly dependent on the data provided by the training set. For example, using the optimal model to predict images downloaded from the Internet is not very accurate. Not using a training set with more data is also a future improvement.

References

[1] World Health Organization. (2005). Fruit and vegetables for health: report of the Joint FAO.

[2] Yang, J., Qiao, P., Li, Y., & Wang, N. (2019). Machine learning classification problems and algorithms, 35(6), 36-40.

[3] Chaganti, S. Y., Nanda, I., Pandi, K. R., Prudhvith, T. G., & Kumar, N. (2020, March). Image Classification using SVM and CNN. In 2020 International Conference on Computer Science, Engineering and Applications (ICCSEA) (pp. 1-5). IEEE.

[4] Baranwal, S. K., Jaiswal, K., Vaibhav, K., Kumar, A., &Srikantaswamy, R. (2020, July). Performance analysis of brain tumour image classification using CNN and SVM. In 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA) (pp. 537-542). IEEE.

[5] Wang, P., Fan, E., & Wang, P. (2021). Comparative analysis of image classification algorithms based on traditional machine learning and deep learning. Pattern Recognition Letters, 141, 61-67.

[6] Zhang, C. (2013). The challenges of machine learning. Science China Press: Information Science, 43(12), 1612-1623.

[7] Xu, B., Cen, K., Huang, J., Shen, H., & Cheng, X. (2020). A Survey on Graph Convolutional Neural Network. Chinese Journal of Computers, 43(5), 755-780.

[8] Sarıgül, M., Ozyildirim, B. M., &Avci, M. (2019). Differential convolutional neural network. Neural Networks, 116, 279-287.

[9] Keras (n.d.). Keras documentation: About keras. Keras. From https://keras.io/about/

[10] Oltean, M. (2021, September 12). Fruits 360. Kaggle. From https://www.kaggle.com/datasets/moltean/fruits

[11] Ge, Y., Liu, H., Wang, Y., Xu, B., Zhou, Q., & Shen, F. (2022). Survey on Deep Learning Image Recognition in Dilemma of Small Samples. Journal of Software, 33(1), 192-210.

[12] Mairal, J., Koniusz, P., Harchaoui, Z., & Schmid, C. (2014). Convolutional kernel networks. Advances in neural information processing systems, 27.

[13] Liu, W., Liang, X., & Qu, H. (2016). Learning performance of convolutional neural networks with different pooling models. Journal of Image and Graphics, 21(9), 1178-1190.

[14] Xie, T. (2021). Research of dropout method based on convolutional neural network. University of Electronic Science and Technology of China.

[15] Lv, F., & Liu, T. (2021). A deep learning algorithm for image recognition with self-enhancement of learning rate. Computer Application and Software, 10.3969.

[16] Deng, J., Zhang, S., Zhang, J., Xun, Y., & Liu, A. (2020). Loss function and application research in supervised learning. Big Data Research, 10.11959.