Hindawi
Security and Communication Networks
Volume 2022, Article ID 8146656, 14 pages
https://doi.org/10.1155/2022/8146656

Research Article

Research on Automatic Cargo Recognition in Smart City Environment

Lanlan Yin, Feng Mo, Qiming Wu, and Zhixun Liang

Hechi University, Yizhou 546300, China

Correspondence should be addressed to Feng Mo; fengmo@hcnu.edu.cn

Received 12 October 2021; Accepted 11 March 2022; Published 18 April 2022

Academic Editor: Gautam Srivastava

Smart city refers to the use of various information technologies to improve the lives of citizens. However, in terms of transportation and sales of goods, traditional methods require a lot of manpower and material resources, and cannot be automatically identified. In order to improve the efficiency and accuracy of product identification, product sorting is automated. It uses the powerful feature learning and expression capabilities of deep convolutional neural networks to automatically learn product features, thereby achieving high-precision image classification. Therefore, this paper first proposes an improved VGG network, combines transfer learning to establish a deep learning recognition model, and finally conducts multiple sets of experiments on the 131-category Fruit-360 dataset. The results show that when the Adam optimizer is used for iterative training for 30 rounds and the batch_size is 64, the accuracy of the algorithm proposed in this paper reaches 94.19% on the training set, 97.91% on the validation set, and 92.2% on the test set top1. The accuracy rate on the test set top5 is as high as 100%. Therefore, the method in this paper can solve the problems caused by traditional methods and provide useful help for smart cities.

1. Introduction

With the advent of the era of big data and artificial intelligence, people's lives have undergone tremendous changes. Information technology has penetrated into all walks of life and has introduced dramatic changes. How to use various information technologies and innovative concepts to connect and integrate urban systems and services, improve the efficiency of resource utilization, optimize urban management and services, and improve the quality of life of citizens is an urgent problem that needs to be solved. Smart city is an advanced form of urban informatization that fully utilizes the new generation of information technology in all walks of life in the city, and realizes the deep integration of informatization, industrialization, and urbanization, which helps alleviate the "big city disease," improve the quality of urbanization, and achieve refinement and dynamic management, and enhance the effectiveness of urban management and improve the quality of life of citizens.

In order to achieve this goal, it is necessary to change many fields with the help of information technology, such as smart transportation, smart logistics, smart agriculture, Internet of Things, Internet of Vehicles, cloud computing, and smart medical care. At present, many excellent scholars have achieved some good research results, including [1–4] for the research on the Internet of Vehicles, and their research can improve the efficiency and safety of urban traffic. [5, 6] Applying deep learning technology to air quality prediction can take measures to solve air problems in advance, which is of great significance to the construction of smart cities. [7] The application of deep learning technology to the problem of sentiment analysis can sense the emotions and psychology of college students in advance, which has become a research hotspot in the fields of psychology, health medicine, and computer science, and has high practical application value.

At present, in the time of artificial intelligence, deep learning technology has penetrated into all walks of life, causing huge changes in the city. Today, it promotes the process of urban informatization, making urban construction gradually move toward digitization, integration, networking, intelligence, and direction development. However,

in people's daily life, there are still traditional manual cargo sorting methods, which cause a lot of resources and labor costs, which runs counter to the construction of smart cities.

In view of the above problems, this paper combines product classification and packaging with deep learning technology to streamline product packaging and automatic distribution, which can greatly reduce the cost of goods loss and personnel sorting. In addition, the application of this technology is of great significance to the whole process of goods distribution, warehousing, warehousing, inventory counting, etc. It changes the way of traditional manual sorting and recording of information, realizes the active perception of logistics information, and greatly simplifies the logistics distribution process, which improves distribution efficiency and is an indispensable link in the construction of smart cities.

Goods recognition and classification based on deep learning have many applications in many fields, such as autonomous navigation, object modeling, process control, or human-computer interaction. We take the fruits and vegetables that are indispensable in people's daily life as examples for identification research. The most interesting application is to create an autonomous robot that can perform more complex tasks than ordinary industrial robots. An example of this is a robot that can perform inspections in the aisles of a store to identify inappropriate items or shelves with insufficient inventory. In addition, the robot can also be enhanced to enable it to interact with the product so that it can solve problems on its own. In addition, this research is of great help to another field of autonomous fruit harvesting.

Although there have been several papers on this topic, as far as we know, they only focus on a few fruits or vegetables. In this article, we are trying to create a network that can classify a variety of fruits and vegetables so that it can be useful in more situations.

We choose to identify fruits and vegetables for several reasons. On the one hand, there are some indistinguishable categories of fruits and vegetables, such as citrus, including oranges and grapefruits. Therefore, we want to see to what extent artificial intelligence can complete the task of classifying them. Another reason is that fruits are common in stores, so they are a good starting point for the aforementioned projects.

In summary, the main contributions of this paper are as follows:

(1) Based on the background of smart city, an intelligent cargo identification method is proposed, which can greatly save resource costs and improve the efficiency of cargo transportation;

(2) The cargo identification method based on artificial intelligence is the starting point of many projects in related fields, and our high-precision identification algorithm can provide favorable support for related fields;

(3) We did not retrain a neural network, but performed partial training with transfer learning technology, which can greatly save hardware costs;

(4) Considering the difficulty and cost of obtaining training data, we use data augmentation to expand the amount of data, which can greatly save time and cost and improve the generalization ability of the model;

(5) By improving the original VGG network, more advanced features are obtained, making the data and network more stable.

2. Related Works

There are mainly two types of target detection models based on deep learning, namely, two-stage target detection models and single-stage target detection models. In 2014, Facebook Artificial Intelligence Laboratory researcher Ross B. Girishick proposed the Region-CNN (R-CNN for short) algorithm [8], which is a two-stage target detection algorithm. The principle of this algorithm is to first extract candidate regions using heuristic algorithms and then perform feature extraction, target classification, and detection on the candidate regions. The R-CNN algorithm applied deep learning technology to target detection for the first time and achieved good results, but at the same time, there are a large number of redundant feature calculations. In 2015, Girishick used the Spatial Pyramid Pooling Network (SPPNet) to propose the Fast R-CNN algorithm, which greatly shortened the running time [9].

In 2015, Ren Shaoqing of Microsoft Research Asia and others proposed the Faster R-CNN algorithm based on Fast R-CNN. This algorithm is based on a convolutional neural network to obtain the feature map of the entire image and replaces it with a custom region suggestion network. The traditional image block extraction algorithm generates the candidate area frame. The feature expression of each candidate area of a fixed length is obtained from the feature map through the method of the region of interest pooling. Finally, the Softmax classifier is used for classification, and the area is obtained through the bounding box regression. The offset of the actual target frame position makes the detected target frame closer to the real position. The innovation is to use the region suggestion network to improve the extraction method of candidate regions, which significantly improves the speed of obtaining candidate regions. In addition, the process of training the network also sets the parameters of the RPN network and the Fast R-CNN network to share the convolutional layer to further improve the learning efficiency of Fast R-CNN and the speed of network detection [10].

In the single-stage target detection algorithm, there is no step of generating candidate regions, and the position size and target category of the frame to be detected are directly predicted, and the detection step is completed at one time. At present, single-stage target detection algorithms can be divided into anchor-based (algorithms) and anchor-free (algorithms). Typical detection models based on anchor points include SSD, YOLOv3, RetinaNet, SqueezeDet, and DetectNet. The YOLO series is a classic single-stage target detection algorithm [11]. This series of algorithms divide the image to be detected into $n \times n$ images of the same size. Each

area corresponds to a certain bounding box. This type of algorithm has fast detection, low background false detection rate, and strong versatility.

At present, many scholars have done some research in the related fields of product identification. Using advanced computer vision, artificial neural network, and PLC control technology, Zhou Wei and others proposed an intelligent control system for fruit classification based on the FX3U-48MT/ES-A PLC controller and analyzed the working principle of the system and the collection and processing of mango samples. With the recognition model, the software and hardware design of the PLC control system is given. The test results show that the fruit classification intelligent control system can use neural network and computer vision methods to properly classify mangoes, with an accuracy of up to 94.23%, which has very important practical significance [12].

Zhu Ling first introduced the idea and principle of the K-means algorithm, then analyzed and studied the acquisition and preprocessing of fruit images, and finally realized the fruit classification and recognition model combining K-means clustering and BP neural network. The test results show that the combination of K-means clustering and BP neural network greatly improves the accuracy of fruit classification and recognition, and greatly shortens the recognition time, which has certain practical significance [13].

Using visual capture technology, Qin National Defence and others have designed a set of automatic fruit sorting systems for picking robots, including conveying mechanism, image acquisition system, control module, and actuators, which can be sorted according to the diameter of apples. The experimental results show that the system classification accuracy rate reached 93.6%, which meets the design requirements, and did not cause any damage to the apple during the classification process, which has a certain degree of effectiveness and reliability [14].

Song proposed a method to identify and count fruits from cluttered greenhouse images [15]. The target plant is pepper, with complex fruit shapes and variable colors, similar to the canopy of plants. The purpose of this application is to locate and count the green and red pepper fruits in large, densely growing pepper plants in the greenhouse. The training and validation data they used included 28,000 images of more than 1,000 plants and their fruits. The pepper positioning and counting method used are two steps: in the first step, the fruit is positioned in a single image, and in the second step, multiple views are combined to improve the detection rate of the fruit. The method of finding pepper fruits in a single image is (1) based on finding points of interest, (2) applying a complex high-dimensional feature descriptor around the points of interest, and (3) using so-called word bags to perform small areas of classification.

Sa proposed a new method to detect fruits from images using deep neural networks [16]. To this end, the author uses a faster region-based convolutional network. Their goal is to create a neural network that can be used by autonomous robots that can harvest fruits. The network uses RGB and near-infrared images for training. The combination of RGB and near-infrared models is completed in two independent situations: early and late fusion. Early fusion means that the input layer has 4 channels: 3 channels for RGB images and 1 channel for near-infrared images. Later fusion uses two independently trained models to merge by obtaining predictions from the two models and averaging the results. The result is a multimodal network with better performance than existing networks.

In the literature [17–19], a fruit detection method based on color, shape, and texture is proposed. They emphasized the difficulty of correctly classifying different kinds of similar fruits. They suggest combining existing methods to detect regions of interest from images using texture, shape, and color. Similarly, in [20], the shape, size, color, texture, and k-nearest neighbor algorithms are combined to improve the accuracy of recognition.

The latest paper [21] proposed an algorithm based on the improved Chan-Vese level-set model, combining the level-set idea and the M-S model [22]. The recommended goal is to conduct night-time green grape testing. Combining the principle of the smallest circumscribed rectangle of the fruit and the Hough line detection method, the picking point is calculated.

In 2021, Richa will use three methods to solve the carrot classification problem [23]. The three methods are KNN, KNN based on cross-validation, and neural network. For the first two methods, the K value needs to be adjusted manually, which will undoubtedly increase the workload. For the third method it uses, it only achieves 77% accuracy on the validation set, which is not very good.

Haq *Z* A studied classification models based on CNN algorithms [24], focusing on the effects of activation functions and convolutional layers on model accuracy and latency. Using a database of 9600 images of three different fruits: apple, banana, and orange, and it can be seen from the simulation that the combination of ReLu-Softmax as an activation function provides the highest percentage increase in accuracy. It can also be seen from the simulation results that ReLu runs the fastest, but has relatively low accuracy for other activation functions.

Mohammad's model detects the open and closed states of pistachios in videos [25]. It is first trained on a RetinaNet network using our dataset to detect different types of pistachios in video frames. Then, after the detections were collected, they were applied to a new counter algorithm based on the new tracker to distribute pistachios in consecutive frames with high accuracy. The algorithm executes very fast and achieves good counting results. Their algorithm achieved a computational accuracy of 94.75% on six videos (9486 frames).

Siddiqi demonstrated how adversarial training improves the robustness of fruit image classifiers [26]. Three convolutional neural network (CNN)-based classifiers IndusNet, fine-tuned VGG16, and fine-tuned MobileNet are proposed. The fine-tuned VGG16 yielded the best test set accuracy of 94.82%, while the other two models were 92.32% and 94.28%, respectively. However, the proposed study also has some limitations. For example, it is still possible to achieve higher accuracy on undisturbed clear images, further reducing overfitting. Also in this study, little preprocessing

was performed on the dataset images, and image preprocessing can be added to the classification process to improve model accuracy.

The following structure of the paper is organized as follows: first, we will describe the Fruit-360 dataset: how it was created and what it contains. Then, we will introduce the principle of the algorithm used in this article and why we chose it. After that, we will introduce in detail the structure of the neural network we use. Next, we will briefly discuss the experimental environment and hardware configuration of this article. Next, the results obtained using the training and test data are described. Finally, we will summarize some improved methods and plans.

3. Materials and Methods

In the field of image recognition and classification, the most successful result is the use of artificial neural networks. These networks form the basis of most deep learning models. Deep learning is a type of machine learning algorithm that uses multilayer nonlinear processing units. Each level learns to transform its input data into a slightly abstract and composite representation.

Deep neural networks have successfully surpassed other machine learning algorithms. They also achieved the first superhuman pattern recognition in some fields. Deep learning is considered to be an important step in gaining powerful artificial intelligence. Second, deep neural networks, especially convolutional neural networks, have achieved good results in the field of image recognition. For this reason, the convolutional neural network is specially applied to the problem of fruit and vegetable recognition.

Next, the rest of this section is organized. We will first introduce the data used by the algorithm, give an overview of transfer learning, introduce the VGG network and its principles, and finally describe the VGG network model based on transfer learning in this article in detail.

3.1. Materials. This article uses an image dataset of popular fruits. The dataset is named Fruit-360 and can be downloaded from the address pointed to by reference [27, 28]. Currently (as of 2020.05.18), the collection contains 90,483 images of 131 types of fruits and vegetables. Each image contains a fruit or vegetable. As shown in Table 1, we use 75% of the data for training, 12.5% for verification, and 12.5% for the final test. The dataset is also available on GitHub and Kaggle, and the original size of the data is $100 \times 100 \times 3$.

3.2. Improved VGG Network Model. Transfer learning is to make the convolutional neural network model trained on a task suitable for a new task through simple adjustments. The convolutional layer of the trained convolutional neural network can perform feature extraction on the image. The extracted feature vector and then input into the fully connected layer with a simple structure can achieve better recognition and classification, so the feature vector extracted by the convolutional layer can be as an image, a more streamlined and more expressive vector. Therefore, the trained convolutional layer plus the fully connected layer suitable for the new task will form a new network model. A little training on the new network model can handle new classification and recognition tasks.

Transfer learning first keeps the structure of the model convolutional layer unchanged and then loads the trained weights and parameters into the convolutional layer. Then, we designed a fully connected layer for the new task and replaced the original fully connected layer with the newly designed fully connected layer to form a new convolutional network model with the original convolutional layer. Finally, use the new image dataset to train the new model. There are two training methods for the new model. One is to freeze the convolutional layer and train only the fully connected layer, and the other is to train all layers of the network.

Convolutional neural network (CNN) is part of the deep learning model. Such a network can be composed of a convolutional layer, a pooling layer, a ReLU layer, a fully connected layer, and a loss layer. In a typical CNN architecture, each convolutional layer is followed by a rectified linear unit (ReLU) layer, and then, a pooling layer is followed by one or more convolutional layers and finally one or more fully connected layers. One feature that distinguishes CNN from ordinary neural networks is that the structure of the image is taken into account when processing the image.

The convolutional layer calculates the input image and the convolution kernel to generate a new feature map. The size of the convolution kernel is generally 3×3 or 5×5. It should be noted that the depth of the input image is the same as the depth of the convolution kernel. Usually, multiple convolution kernels of different sizes can be extracted from the input image to obtain different feature maps.

Assuming that the width and height of the input image are W_{input}, W_{input}, and H_{input}, H_{input}; the width and height of the convolution kernel are W_{filter}, W_{filter}, and H_{filter}, H_{filter}; the step size is S; and the padding is P, and then, the width and height (W_{out}, W_{out}, H_{out}, H_{out}) of the resulting feature map are defined by the following calculation formulas:

$$\begin{aligned} \text{W}_{\text{out}} &= \frac{W_{\text{input}} - W_{\text{filter}} + 2P}{S} + 1, \\ \text{H}_{\text{out}} &= \frac{H_{\text{input}} - H_{\text{filter}} + 2P}{S} + 1. \end{aligned} \tag{1}$$

The function of the pooling layer is to reduce the size of the model, increase the calculation speed, and also improve the robustness of the extracted features. There are two types of pooling operations, namely, maximum pooling and average pooling.

Assuming that the width and height of the input image are Winput and H_{input}, the width and height of the convolution kernel are W_{filter} and H_{filter}, the step size is S, and the padding is P, the calculation formulas for the width and the height (W_{out}, H_{out}) of the obtained feature map are as follows:

$$\begin{aligned} \text{W}_{\text{out}} &= \frac{W_{\text{input}} - W_{\text{filter}}}{S} + 1, \\ \text{H}_{\text{out}} &= \frac{H_{\text{input}} - H_{\text{filter}}}{S} + 1. \end{aligned} \tag{2}$$

Table 1: Number of images for each fruit.

Category	n_train	n_valid	n_test
Grape blue	984	164	164
Plum 3	900	152	152
Peach 2	738	123	123
Strawberry wedge	738	123	123
Tomato 1	738	123	123
Melon Piel de Sapo	738	123	123
Tomato 3	738	123	123
Cherry rainier	738	123	123
Cherry 2	738	123	123
Walnut	735	124	125
Pear stone	711	118	119
Pepper orange	702	117	117
Cauliflower	702	117	117
Fig	702	117	117
Pear Forelle	702	117	117
Pear 2	696	116	116
Tomato heart	684	114	114
Tomato 2	672	112	113
Apple red yellow 2	672	109	110
Pepper yellow	666	111	111
Pear red	666	111	111
Pepper red	666	111	111
Nut forest	654	109	109
Nut pecan	534	89	89
Pineapple mini	493	81	82
Rambutan	492	82	82
Grape pink	492	82	82
Grape white 3	492	82	82
Grapefruit white	492	82	82
Physalis	492	82	82
Lemon	492	82	82
Pomegranate	492	82	82
Pear	492	82	82
Peach flat	492	82	82
Peach	492	82	82
Papaya	492	82	82
Mulberry	492	82	82
Nectarine	492	82	82
Physalis with husk	492	82	82
Redcurrant	492	82	82
Apple braeburn	492	82	82
Apple red yellow 1	492	82	82
Apple red 2	492	82	82
Cantaloupe 1	492	82	82
Cherry 1	492	82	82
Cherry wax black	492	82	82
Cherry wax red	492	82	82
Tomato cherry red	492	82	82
Apricot	492	82	82
Cherry wax yellow	492	82	82
Cantaloupe 2	492	82	82
Apple red 1	492	82	82
Apple granny smith	492	82	82
Strawberry	492	82	82
Apple golden 2	492	82	82
Avocado ripe	491	83	83
Pear abate	490	83	83
Pineapple	490	83	83
Pepino	490	83	83
Cactus fruit	490	83	83

Table 1: Continued.

Category	n_train	n_valid	n_test
Banana red	490	83	83
Pear Williams	490	83	83
Banana	490	83	83
Apple red delicious	490	83	83
Passion fruit	490	83	83
Pear monster	490	83	83
Dates	490	83	83
Salak	490	81	81
Carambola	490	83	83
Maracuja	490	83	83
Kaki	490	83	83
Granadilla	490	83	83
Tamarillo	490	83	83
Grape white	490	83	83
Tangelo	490	83	83
Cocos	490	83	83
Raspberry	490	83	83
Grapefruit pink	490	83	83
Clementine	490	83	83
Guava	490	83	83
Pitahaya red	490	83	83
Huckleberry	490	83	83
Quince	490	83	83
Kumquats	490	83	83
Meyer Lemon	490	83	83
Limes	490	83	83
Lychee	490	83	83
Mandarine	490	83	83
Mango	490	83	83
Grape white 2	490	83	83
Apple golden 3	481	80	81
Nectarine flat	480	80	80
Apple golden 1	480	80	80
Orange	479	80	80
Tomato 4	479	80	80
Watermelon	475	78	79
Tomato not ripened	474	79	79
Grape white 4	471	79	79
Kohlrabi	471	78	79
Cucumber ripe 2	468	78	78
Eggplant	468	78	78
Kiwi	466	78	78
Hazelnut	464	78	79
Blueberry	462	77	77
Corn husk	462	77	77
Tomato yellow	459	76	77
Apple pink lady	456	76	76
Potato red washed	453	75	76
Banana lady finger	450	76	76
Pomelo sweetie	450	76	77
Corn	450	75	75
Potato sweet	450	75	75
Potato white	450	75	75
Beetroot	450	75	75
Onion red	450	75	75
Chestnut	450	76	77
Potato red	450	75	75
Plum	447	75	76
Onion red peeled	445	77	78
Pepper green	444	74	74

Table 1: Continued.

Category	n_train	n_valid	n_test
Apple crimson snow	444	74	74
Onion white	438	73	73
Apple red 3	429	72	72
Avocado	427	71	72
Mango red	426	71	71
Plum 2	420	71	71
Cucumber ripe	392	65	65
Tomato maroon	367	63	64
Pear kaiser	300	51	51
Mangosteen	300	51	51
Ginger root	297	49	50

As the last layer or multiple layers of the neural network, the fully connected layer plays a role in feature space transformation and dimensionality reduction. It can transform the feature transformation of the previous layer into a new feature space and convert high-dimensional features into one-dimensional features, which is convenient for the final classification prediction of the model.

The classic representative of the convolutional neural network is the VGG16 network, which is composed of 13 convolution modules and 3 fully connected modules. The output number of the last fully connected layer is 1000, corresponding to the number of target categories, and SoftMax is used to calculate the loss, as shown in Figure 1.

VGG-16 completed training on the ImageNet dataset, and the training set alone reached 1.28 million. The amount of data and the number of sample types are enough to get a model with strong expressive ability. However, there is currently no large enough dataset for fruit images, and considering the high training cost, it is difficult to train the network model to the ideal classification effect. Therefore, the method of transfer learning can be used to realize the task of fruit and vegetable classification. We retain the model structure of the first 13 layers in Figure 1 and then redesign the fully connected module. The improved fully connected module is shown in Figure 2.

The input image can be converted into a $7 \times 7 \times 512$ three-dimensional vector after extracting features in the first 13 layers of VGG16, and the dimension is reduced to 1×4096 through the fully connected layer 1. After entering the nonlinear activation function ReLU, the model uses the ReLU activation function. The ReLU function has the characteristics of simple calculation and fast convergence, and its expression is as follows:

$$\text{Re}lu = f(x) = \max(0, x). \tag{3}$$

Then, we enter the dropout layer [29]. The dropout layer temporarily sets the weight of some neurons to 0.5 according to a certain probability during each training process of the network, which can alleviate the coordinated adaptation between neurons and reduce the dependence between neurons, avoid overfitting of the network, and then enter the fully connected layer 2 to further reduce the dimension of the vector to 1×4096.

After that, the ReLU layer is also subjected to nonlinear transformation, and then, into the dropout layer, some neurons are disabled according to the probability of 0.5. Then proceed to the third fully connected layer to reduce the dimensionality of the feature to a one-dimensional vector of 1×256.

Then, the 1×256 features are activated by ReLU, enter the dropout layer to inactivate some neurons with a probability of 0.4, and then reduce the dimensionality to 1×131. Finally, LogSoftmax is performed to calculate the scoring probability of each category. Softmax is not used here because it compresses the value to (0, 1), while the value range of LogSoftmax is $(-\infty, 0)$. This can prevent overflow problems and facilitate the calculation of the loss function.

$$\begin{aligned} \text{Softmax} &= \sigma(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{131} e^{z_i}}, \\ \text{LogSoftmax} &= \log(\sigma(z_i)) = \log\left(\frac{e^{z_i}}{\sum_{j=1}^{131} e^{z_i}}\right). \end{aligned} \tag{4}$$

In summary, the model structure in this section is shown in Figure 3 below. First, we migrate the model parameters of the classic VGG16 network on ImageNet to our VGG16 network, while freezing the first 16 layers of VGG16 and changing the last layer of VGG16. The number of classifications in the connection layer is 131, and then, LogSoftmax is used for classification prediction and verification, and finally tested through the test set.

Based on the overview diagram in Figure 3, the model in this paper can be further divided into a feature extraction part and a classifier part, as shown in Tables 2 and 3 respectively. The input image is a $100 \times 100 \times 3$ image, and the data become $224 \times 224 \times 3$ after data enhancement as the input of the model. After that, feature extraction is performed according to the parameters in Table 2. After each convolution, there is a ReLu nonlinear operation. Because it does not change the input and output sizes, it is not reflected in Table 3.

As shown in Table 3, the feature in Table 2 is first subjected to a linear operation into a 4096-dimensional vector, and then, a 50% dropout operation is performed, and so on. Until the 131-dimensional column vector is finally obtained, the final classification result is obtained through LogSoftmax operation.

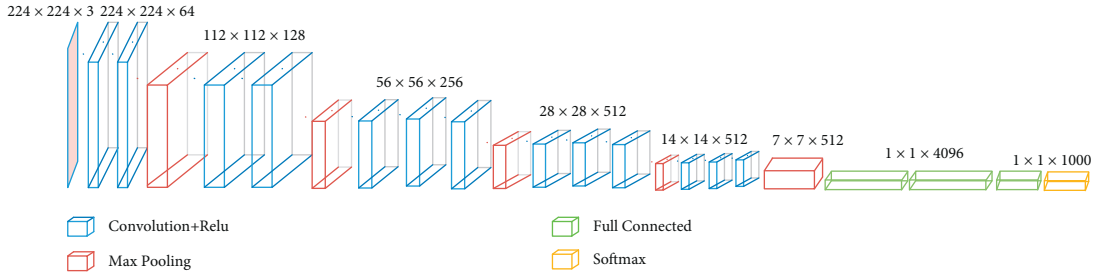

Figure 1: Traditional VGG16 network structure.

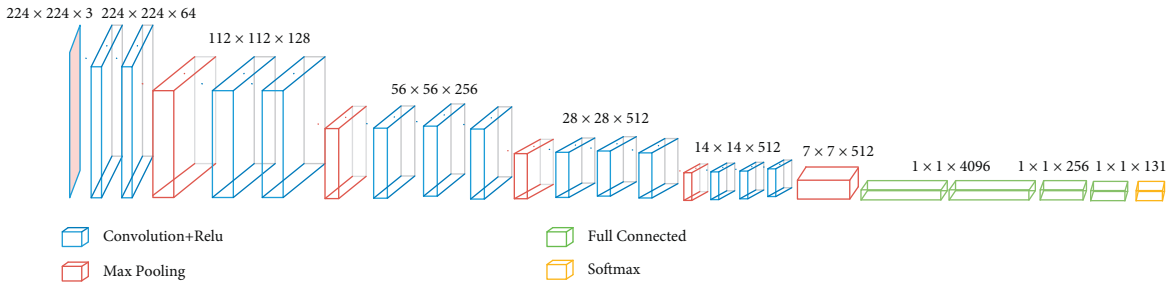

Figure 2: The improved structure of VGG16 in this article.

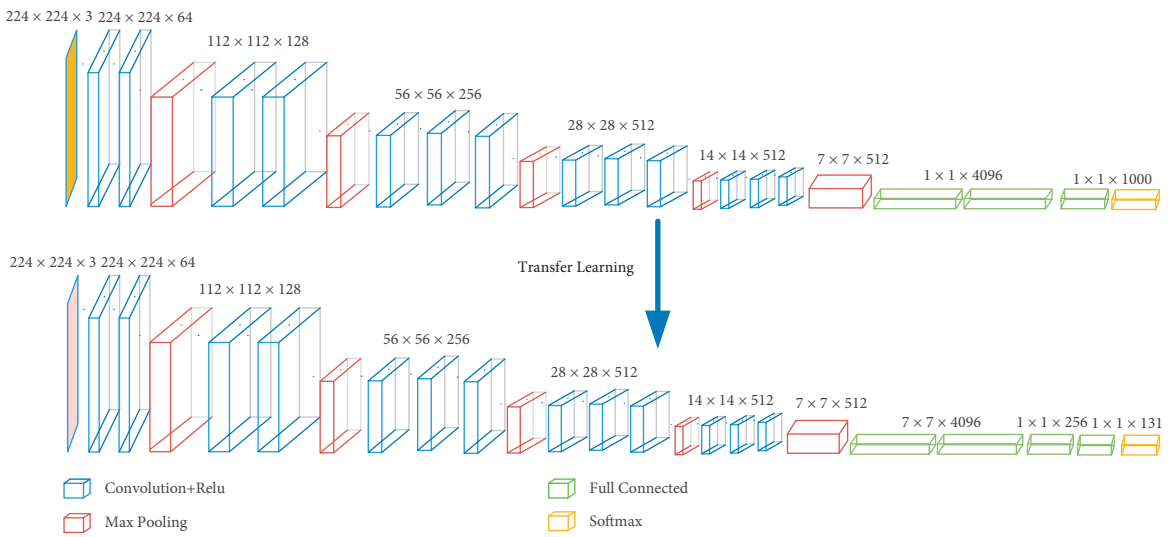

Figure 3: Overview of the network structure mentioned in this article.

4. Results and Discussion

4.1. Experimental Environment. The experiment was completed in Python 3, PyTorch 1.7, and CUDA 10.2 software environment. In the hardware environment, the CPU adopts Intel core i7-10875H, the main frequency is 2.3 GHz eight-core, and the GPU adopts Nvidia GeForce RTX 2070 Super, 8 GB video memory.

4.2. Experimental Results and Analysis. Due to the limited number of images in certain categories, we first use image enhancement to artificially increase the number of images "see" by the network. This means that for the training set, we will randomly adjust, crop, and flip the image horizontally to increase the dataset. Each stage (during training) applies a different random transformation, so the network effectively sees many different versions of the same image. All data are also converted to Torch tensor before normalization. The training set enhancement methods and enhancement effects used in the experiment are shown in Table 4 and Figure 4, respectively. The validation and test data are not augmented, just resized, and normalized, as shown in Table 5. It can be seen that the same image in the training set has been transformed into 16 different images after data enhancement, and this operation can expand the data by 16 times.

Then Batch_size selects 64, and the optimizer selects Adam. The final experiment achieves 94.19% accuracy on the

Table 2: The layout of the features part.

Type	Kernels/Kernel_size	Stride	Input/output
Con2d	64/3 × 3	1	224 × 224 × 3/224 × 224 × 64
Con2d	64/3 × 3	1	224 × 224 × 64/224 × 224 × 64
MaxPool2d	2 × 2	2	224 × 224 × 64/112 × 112 × 64
Con2d	128/3 × 3	1	112 × 112 × 64/112 × 112 × 128
Con2d	128/3 × 3	1	112 × 112 × 128/112 × 112 × 128
MaxPool2d	2 × 2	2	112 × 112 × 128/56 × 56 × 128
Con2d	256/3 × 3	1	56 × 56 × 128/56 × 56 × 256
Con2d	256/3 × 3	1	56 × 56 × 256/56 × 56 × 256
Con2d	256/3 × 3	1	56 × 56 × 256/56 × 56 × 256
MaxPool2d	2 × 2	2	56 × 56 × 256/28 × 28 × 256
Con2d	512/3 × 3	1	28 × 28 × 256/28 × 28 × 512
Con2d	512/3 × 3	1	28 × 28 × 512/28 × 28 × 512
Con2d	512/3 × 3	1	28 × 28 × 512/28 × 28 × 512
MaxPool2d	2 × 2	2	28 × 28 × 512/14 × 14 × 512
Con2d	512/3 × 3	1	14 × 14 × 512/14 × 14 × 512
Con2d	512/3 × 3	1	14 × 14 × 512/14 × 14 × 512
Con2d	512/3 × 3	1	14 × 14 × 512/14 × 14 × 512
MaxPool2d	2 × 2	2	14 × 14 × 512/7 × 7 × 512

Table 3: The layout of the classifier part.

Type	Parameters
Linear	(25088, 4096)
Dropout	0.5
Linear	(4096, 4096)
Dropout	0.5
Linear	(4096, 256)
Dropout	0.4
Linear	(256, 131)
LogSoftmax	

Table 4: Training data augmentation.

Augmentation	Setting
RandomResizedCrop	size = 256, scale = (0.8, 1.0))
RandomRotation	degrees = 15
CenterCrop	size = 224
Normalize	Mean=([0.485, 0.456, 0.406], std = [0.229, 0.224, 0.225])

training set, 98.46% accuracy on the validation set, 93% accuracy on the test set top1, and even more on the test set top5 reaching 100% accuracy rate. Figure 5 below is the error change of the training set and the validation set for 30 rounds of training, and Figure 6 is the accuracy change curve of the training set and the validation set.

Next, the model is tested, and the results of the test set are as follows. As shown in Figures 7 and 8, the accuracy of the top 5 fruits and vegetables such as redcurrant and apricot can reach 100%.

Then, the 131 types of fruits and vegetables in the training set were tested. The average accuracy of top1 reached 92.2, and the average accuracy of top5 reached 100%. The experimental results of some fruits and vegetables are shown in Table 1. It can be seen that on the five fruits and vegetables in Table 6, the model proposed in this article can achieve an accuracy of more than 98%.

Finally, verify the relationship between the number of fruit and vegetable images in the training set and the accuracy of top1. As shown in Figure 9, it can be seen that the number of random images of the model in this paper is increasing, and the accuracy is gradually improving, basically above 90%, and gradually approaching 100%. Figure 10 verifies the relationship between the number of images of fruits and vegetables in the training set and the accuracy of top5. It can be seen that as the number of images increases, the effect of the model in this paper is relatively stable, all at 100%.

Figure 4: Data enhancement results.

Table 5: Val or Test datasets.

Augmentation	Setting
Resize	size = 256,
CenterCrop	size = 224
Normalize	Mean = ([0.485, 0.456, 0.406], std = [0.229, 0.224, 0.225])

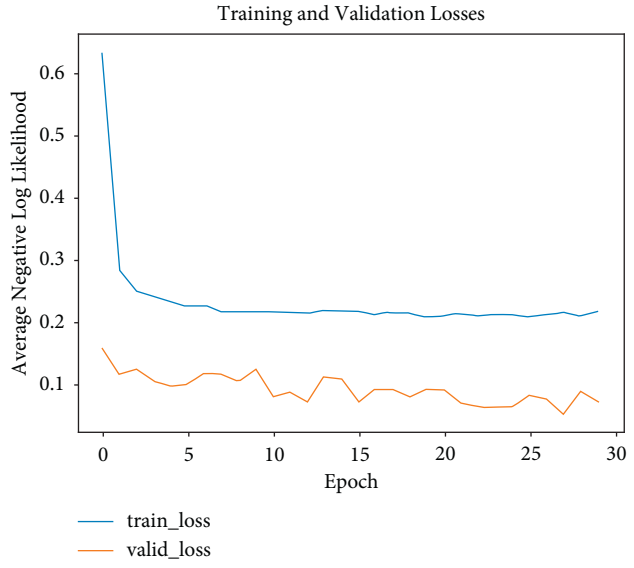

Figure 5: Error curve of training set and validation set.

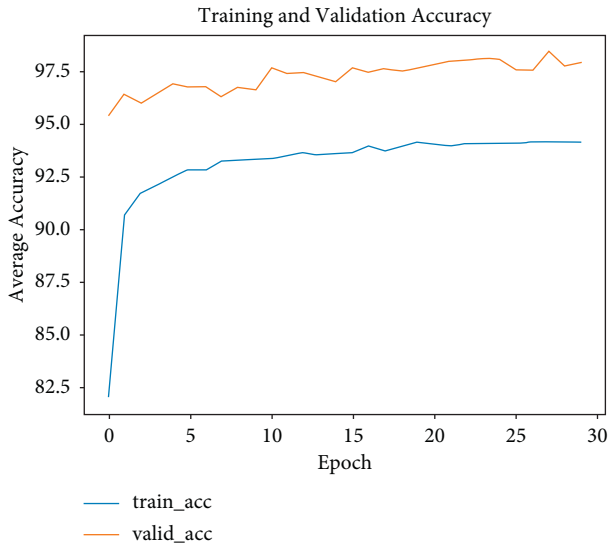

Figure 6: Accuracy of the training set and validation set.

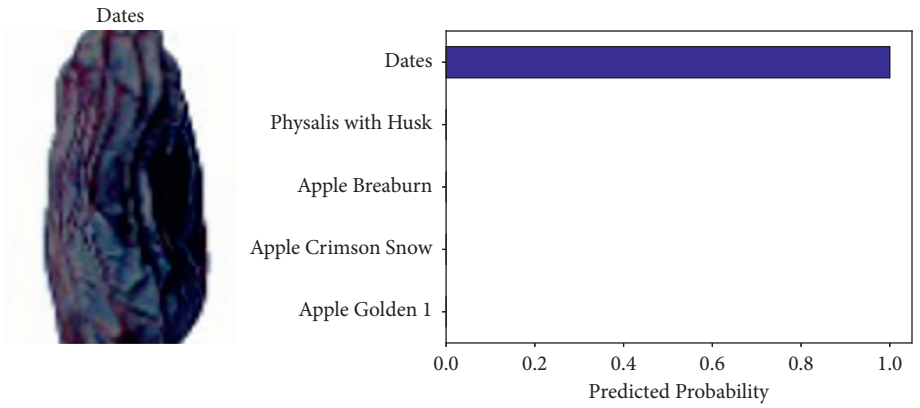

Figure 7: Dates top5 accuracy rate.

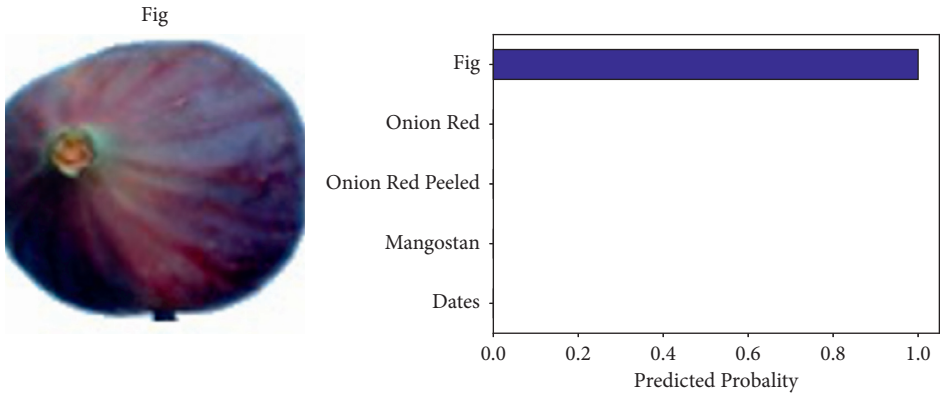

Figure 8: Fig top5 accuracy rate.

Table 6: Part of the test results in the test data.

Class	top1 (%)	top5 (%)	Loss
Apple braeburn	100.0	100.0	$1.5e-02$
Apple crimson snow	98.7	100.0	$7.9e-02$
Apple golden 1	100.0	100.0	$3.2e-07$
Apple golden 2	100.0	100.0	$3.7e-03$
Apple golden 3	84.0	100.0	$4.9e-01$

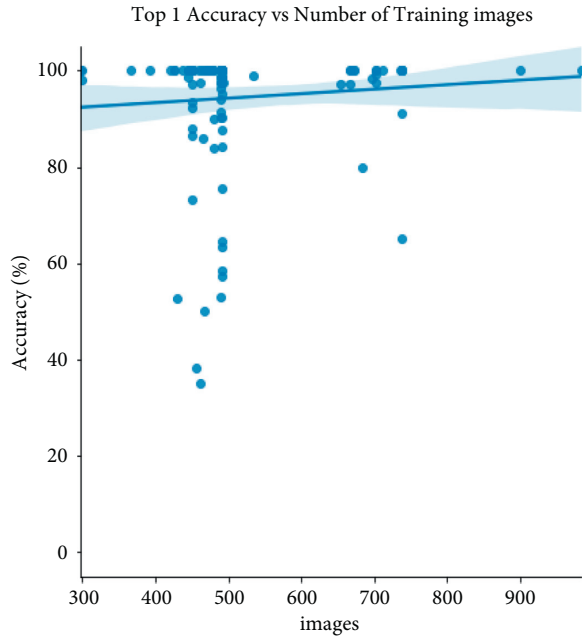

Figure 9: The number and accuracy of top1 in the training set.

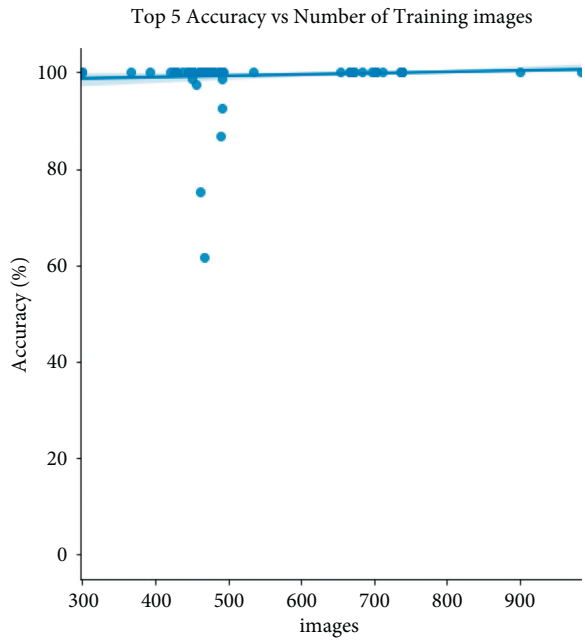

Figure 10: The number of images in the training set and the accuracy of the top5.

5. Conclusion

Aiming at the time-consuming and labor-intensive problems of traditional fruit and vegetable sorting and identification, a multicategory fruit and vegetable identification model based on transfer learning are proposed. First, in order to ensure the diversity of data and improve the generalization ability of the model, we performed data enhancement on the original data and performed random adjustment, cropping, and horizontal flipping operations, respectively. This allows the network to see more different data and improve the recognition rate of the model.

Second, in order to save the training cost, the parameters on the ImageNet dataset are transferred to the improved VGG network model in this paper, which speeds up the training time. Because we freeze the first 12 layers of VGG16, the network parameters of the first 12 layers have been trained on the ImageNet dataset for multiple rounds, and only the last few layers are mainly trained, which can ensure that there is a relatively high level at the beginning of training (recognition accuracy).

Finally, the classic VGG16 network is improved, the last 1000-dimensional fully connected layer and Softmax layer are deleted, two layers of 256-dimensional and 131-dimensional fully connected layers are added, and the LogSoftmax function is used to replace Softmax because we believe that deeper networks will extract higher-level features, which will ensure higher recognition accuracy. Compared with Softmax, LogSoftmax will better ensure the stability of data and prevent overflow.

The research presented in this paper also has some limitations. These limitations could form the basis of future research work. The following are the identified constraints and related directions for future work:

(1) This paper directly conducts classification experiments on 131 types of fruits, but there are inevitably errors in identification, because the gap between some fruits is extremely small. Apple, for example, has 13 different types of subcategories. Next, you can first perform the division prediction of large categories and then perform the prediction of subcategories under the same category, which may have a better recognition effect.

(2) Considering the hardware and computing costs, this paper does not train a brand new neural network from 0, which can be used as the next method to improve the classification accuracy.

(3) The work in this paper only considers the recognition and classification work under the single-target situation and does not consider the training work under the multiobjective situation. In practical scenarios, the recognition of multiobjective tasks may be more general and more meaningful.

(4) In the future, considering the problems of model landing and deployment, the model should be miniaturized to improve the model recognition accuracy and operation efficiency.

Data Availability

The public dataset can be obtained from https://github.com/Horea94/Fruit-Images-Dataset/.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The study was supported by 2019 Guangxi Basic Research Ability Improvement Project for Young and Middle-Aged University Teachers (2019KY0640).

References

[1] X. Xu, Z. Fang, J. Zhang et al., "Edge content caching with deep spatiotemporal residual network for IoV in smart city," *ACM Transactions on Sensor Networks*, vol. 17, no. 3, pp. 1–33, 2021.

[2] X. Xu, Z. Fang, L. Qi, X. Zhang, Q. He, and X. Zhou, "TripRes," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 17, no. 2, pp. 1–21, 2021.

[3] B. Shen, X. Xu, L. Qi, X. Zhang, and G. Srivastava, "Dynamic server placement in edge computing toward internet of vehicles," *Computer Communications*, vol. 178, pp. 114–123, 2021.

[4] X. Xu, H. Li, W. Xu, Z. Liu, L. Yao, and F. Dai, "Artificial intelligence for edge service optimization in internet of vehicles: a survey," *Tsinghua Science and Technology*, vol. 27, no. 2, pp. 270–287, 2022.

[5] Z. Hu, X. Xu, Y. Zhang et al., "Cloud-edge cooperation for meteorological radar big data: a review of data quality control–edge cooperation for meteorological radar big data: a review of data quality control," *Complex & Intelligent Systems*, pp. 1–15, 2021.

[6] W. Kong and B. Wang, "Combining trend-based loss with neural network for air quality forecasting in internet of Things," *Computer Modeling in Engineering and Sciences*, vol. 125, no. 2, pp. 849–863, 2020.

[7] X. Lu and H. Zhang, "An emotion analysis method using multi-channel convolution neural network in social networks," *Computer Modeling in Engineering and Sciences*, vol. 125, no. 1, pp. 281–297, 2020.

[8] R. Girshick, J. Donahue, and T. Darrell, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580–587, IEEE, Columbus, Ohio, November, 2014.

[9] R. Girshick, "Fast R-CNN," in *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 1440–1448, Santiago, Chile, June, 2015.

[10] S. Q. Ren, K. M. He, and R. Girshick, "Faster R-CNN: towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, pp. 1137–1149, 2015.

[11] J. Redmon, S. Divvala, and R. Girshick, "You only look once: unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016*, pp. 779–788, IEEE, USA, June, 2016.

[12] W. Zhou, X. Yingruo, Intelligent control system for fruit classification based on PLC and image processing," *Journal of Agricultural Mechanization Research*, vol. 43, no. 05, pp. 235–239, 2021.

[13] L. Zhu, "Recognition and application of fruit classification based on K - means clustering algorithms," *Journal of Agricultural Mechanization Research*, vol. 42, no. 08, pp. 46–50, 2020.

[14] G. Qin and M. Qin, "Application of visual capture picking robot in fruit classification system," *Journal of Agricultural Mechanization Research*, vol. 42, no. 09, pp. 212–216, 2020.

[15] Y. Song, C. A. Glasbey, G. W. Horgan, G. Polder, J. A. Dieleman, and G. W. A. M. van der Heijden, "Automatic fruit recognition and counting from multiple images," *Biosystems Engineering*, vol. 118, pp. 203–215, 2014.

[16] I. Sa, Z. Ge, F. Dayoub, B. Upcroft, T. Perez, and C. McCool, "Deepfruits: a fruit detection system using deep neural networks," *Sensors*, vol. 16, p. 8, 2016.

[17] A. Selvaraj, N. Shebiah, S. Nidhyananthan, and L. Ganesan, "Fruit recognition using color and texture features," *Journal of Emerging TRends in Computing and Information Sciences*, vol. 1, no. 10, pp. 90–94, 2010.

[18] H. Zawbaa, M. Abbass, M. Hazman, and A. E. andHassanien, "Automatic fruit image recognition system based on shape and color features," *Communications in Computer and Information Science*, vol. 488, no. 11, pp. 278–290, 2014.

[19] D. Li, H. Zhao, X. Zhao, Q. Gao, and L. Xu, "Cucumber detection based on texture and color in greenhouse," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 31, no. 01, 2017.

[20] R. K. Srivastava, K. Greff, and J. Schmidhuber, "Training very deep networks," *CoRR abs/*, vol. 1507, p. 06228, 2015.

[21] J. Xiong, Z. Liu, R. Lin et al., "Green grape detection and picking-point calculation in a night-time natural environment using a charge-coupled device (ccd) vision sensor with artificial illumination," *Sensors*, vol. 18, p. 4, 2018.

[22] D. Mumford and J. Shah, "Optimal approximations by piecewise smooth functions and associated variational problems," *Communications on Pure and Applied Mathematics*, vol. 42, no. 5, pp. 577–685, 1989.

[23] R. Sharma, A. Agarwal, and H. R. Mamatha, "Classification of carrots based on shape analysis using machine learning techniques," in *Proceedings of the 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)*, pp. 1407–1411, IEEE, Tirunelveli, India, February, 2021.

[24] Z. A. Haq and Z. A. Jaffery, "Impact of activation functions and number of layers on the classification of fruits using CNN," in *Proceedings of the 2021 8th International Conference on Computing for Sustainable Global Development (INDIACom)*, pp. 227–231, IEEE, New Delhi, India, March, 2021.

[25] M. Rahimzadeh and A. Attar, "Detecting and counting pistachios based on deep learning," *Iran Journal of Computer Science*, vol. 5, pp. 1–13, 2021.
[26] R. Siddiqi, "Fruit-classification model resilience under adversarial attack," *SN Applied Sciences*, vol. 4, no. 1, pp. 1–22, 2022.
[27] D. C. Ciresan, U. Meier, J. Masci, L. M. Gambardella, and J. F. Schmid-huber, "High performance convolutional neural networks for image classification," in *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence*, pp. 1237–1242, AAAI Press, Barcelona, Catalonia, Spain, July, 2011.
[28] H. Muresan and M. Oltean, "Fruit recognition from images using deep learning," *Acta Universitatis Sapientiae, Informatica*, vol. 10, pp. 26–42, 2018.
[29] J. Schmidhuber, "Deep learning in neural networks: an overview," *CoRR abs/*, vol. 1404, p. 7828, 2014.