Journal of Network Intelligence
Taiwan Ubiquitous Information

Volume **7**, Number **2**, May **2022**

Recognition Method and Application of Wild Vegetables based on Lightweight Convolutional Neural Network Model

Bingbing Wang

College of Information and Electrical Engineering
China Agricultural University
Key Laboratory of Agricultural Informatization Standardization
Ministry of Agriculture and Rural Affairs
17 Tsinghua East Road, Haidian, Beijing
snowice310@163.com

Wanlin Gao*

College of Information and Electrical Engineering
China Agricultural University
Key Laboratory of Agricultural Informatization Standardization
Ministry of Agriculture and Rural Affairs
17 Tsinghua East Road, Haidian, Beijing
Corresponding author: wanlingaocau@163.com

Bangjie Yang

College of Information and Electrical Engineering
China Agricultural University
Key Laboratory of Agricultural Informatization Standardization
Ministry of Agriculture and Rural Affairs
17 Tsinghua East Road, Haidian, Beijing
bjyang@agri.gov.cn

Received October 2021; Revised December 2021

Abstract. *Wild vegetables are widely eaten by people for its homology of medicine and food. The identification of edible wild vegetables has been mainly based on manual and empirical identification, which can lead to low recognition accuracy. To improve the identification efficiency and accuracy of wild vegetables, computer vision and deep learning technologies were used to build a wild vegetable recognition and classification model, and based on this model, an Android-based App for wild vegetables recognition and classification was developed. 5 common convolutional neural network (CNN) models of MobileNetV1, MobileNetV2, VGG-16, Google Net and InceptionV3 were used to train wild vegetables datasets with 10 classes. The results show that the model of MobileNetV2 and VGG-16 have higher recognition accuracy, which is 96.92% and 95.31% respectively. According to the recognition accuracy and size of the model, the model of MobileNetV2 was choose to be transplanted to the Android mobile phone. After the model is migrated to the mobile phone, the running memory is 28.89Mb, and the average running time is 418ms, which is more suitable for mobile phone development than other models. The App developed in this paper can better realize recognition and classification of wild vegetables, and facilitate the development and utilization of wild vegetable resources.*

Keywords: Wild Vegetable identification, Convolutional neural network, Wild vegetable identification App

---

1. **Introduction.** Wild vegetables refer to wild or semi wild plants that grow naturally in the field and have no artificial cultivation, and their roots, stems, leaves, flowers, fruits, and other organs can be used as vegetables [1–3]. People who have knowledge of wild vegetables are elderly people and women in China. Their knowledge is less comprehensive, and most of them rely on empirical judgments. Moreover, insufficient understanding of specific nutritional components, medical and health care functions, and unscientific methods can easily lead to wild vegetable poisoning incidents [4]. In recent years, with the development of computer technology and pattern recognition technology, it has become possible to use computers and smartphones to identify and classify plants. It greatly improves the efficiency of identification and classification with the characteristics of accuracy, fast, ease of use. And then, it overcomes the time-consuming, labor-intensive, and inefficient problems of traditional manual methods. For this reason, it is necessary for us to use computer technology to develop a wild vegetables recognition App.
Nowadays, deep learning has been widely used in plant recognition and classification. More and more recognition models have been applied to mobile platforms. Compared with the traditional image recognition technology, deep learning reduces the complex feature extraction engineering and solves the difficulties of image recognition in low feature extraction efficiency, low recognition accuracy and slow speed. CNN shows great advantages in image recognition and classification,

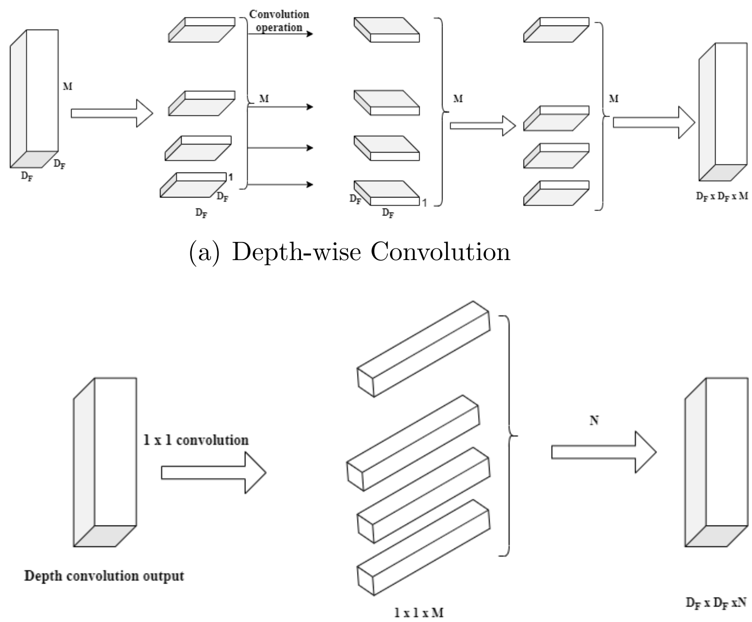
and is more and more used in plant recognition and classification research, plant organ recognition and detection [5–9]. In 2017, Zhang Xueqin et al. proposed a plant recognition algorithm based on P-AlexNet model. He optimized the deep learning model based on AlexNet by transfer learning, and improved the generalization ability, representation ability of detail features and recognition accuracy of the model [10]. Zhang Jianhua et al. optimized the fully connected layers based on original VGG-16 network model. He proposed an improved VGG CNN plant leaf recognition model, and the recognition effect reached 89.51% [11]. Pierre Barré et al. constructed a Leaf-Net CNN model for automatic identification of plant leaves, and conducted experimental verification on several common leaf data sets. The results show that this model is better than traditional manual features in plant leaf automatic identification. The accuracy and efficiency of extracting classification and recognition models have been improved [12]. MadsDyrmann et al. used deep CNNs to build a model for identifying and classifying 22 plant pictures under different illumination, resolution, and soil types, with a recognition accuracy of 86.2% [13]. In addition, AzeddineElhassouny et al. constructed a smart phone application model for tomato leaf disease recognition based on deep CNNs. This model can identify 10 common tomato leaf diseases with high diagnostic accuracy [14]. Thi et al. proposed an automatic plant image recognition system, which uses a new method of plant image recognition based on the combination of deep learning, migration learning and crawling technology, and developed a Vietnamese plant recognition retrieval system [15].

According to these researches, the recognition accuracy and functions of these plant classification and recognition models and systems are different, and the application scenarios and model generalization capabilities are weak. Model training data sets mostly use leaf data with a single background as the object, and there are fewer network models for classification of complex background images. These models have many parameters and occupies a large of computing memory, which is not suitable for deployment in mobile terminals such as mobile phones [16].

In this study, the related work was described in the second section. In the third section, we introduced the methods that used when build the identification model. And images processing technology and transfer learning method were used during establish the recognition and classification model. In the next, the experiment results were described. Then the recognition accuracy, model size, and calculation speed of the five classification models are analyzed and compared. Finally, a network model that is more comprehensively suitable for mobile terminal is selected to transplant the models and develop a wild vegetable recognition App by using the smart phone development platform Android Studio.

2. **Related work.**

2.1. **MobileNet Convolutional Neural Network.** MobileNet is a lightweight CNN launched by Google in 2017. It aims to make full use of the limited resources of mobile devices and embedded applications, under the premise of ensuring high network accuracy, to reduce channels, convolution core size and disk usage. Through the removal of some feature maps, pruning and discretizing some unimportant weights, the network complexity is reduced to meet various application cases under limited resources [17]. MobileNet can be used for image feature extraction for tasks such as classification, detection, embedding and segmentation like convolutional networks, such as VGG and ResNet [18–21]. The core unit of MobileNet is depth-wise separable convolution, which is an operation that decomposes standard convolution into depth-wise convolution and pointwise convolution, as shown in Figure 1.



(a) Depth-wise Convolution

(b) Point-wise Convolution

FIGURE 1. Schematic diagram of depth-wise convolution and point-wise convolution. DF is the length and width of the input; M is the input channels and convolution kernels, and the output size is the same as the input. 11 is the size of the convolution kernel, and N is the number of output channels.

Different from the traditional standard convolution, the depth separable convolution is that: first perform the depth-wise convolution, and use the filter to convolve each input channel; then use the point-by-point convolution and the result calculated by depth-wise convolution to perform a 1×1 convolution kernel operation, and obtain the output result. Compared with the traditional standard convolution, this calculation method can effectively reduce the calculation and model parameter. Suppose the input feature map is $D_F * D_F * M$, and the output is $D_F * D_F * N$. According to the standard convolution calculation method, the required size is $D_K * D_K$ is the length of the convolution kernel, ant the calculation amount is $D_K * D_K * M * N * D_F * D_F$. The calculation of the depth separable convolution is the calculation sum of the depth convolution and the point-wise convolution. As shown in Figure 1(a), the deep convolution first decomposes the input feature map into M, and then performs the convolution operation on each channel separately to obtain M features. Finally, M features are combined to obtain the output result, and the calculation amount of the deep convolution is $D_K * D_K * M * D_F * D_F$. Since the output of the deep convolution is a point-by-point input, it can be seen from Figure 1(b) that perform a $1 * 1$ convolution and merge operation to the input, the calculation amount of the point-by-point convolution is $M * D_F * D_F * N$.Compared with the traditional convolution, the depth separable convolution method is:

$$\frac{D_K * D_K * M * D_F * D_F + M * N * D_F * D_F}{D_F * D_F * M * N * D_F * D_F} = \frac{1}{N} + \frac{1}{D_K^2} \tag{1}$$

2.2. **Data augmentation technology.** Data augmentation is a technology that generates more useful new data from existing data to expand the capacity of data set. The data set amplification is mainly to reduce the over-fitting of network. By transforming the training pictures, a network with stronger generalization ability can be obtained, which can better adapt to the application scenario [22]. The current data augmentation techniques in deep learning include:

1) Translation transformation: A method of generating a new image by moving all pixels in the original image in a certain direction and moving a certain distance in parallel.

2) Rotation transformation: A method of generating another image by rotating the original image at a certain angle. During the rotation, all pixels of the original image are rotated by the same angle in the same direction.

3) Flip transformation: It is also called specular reflection transformation or axisymmetric transformation. Normally, only horizontal and vertical inversion transformations are used for the original image.

4) Scaling transformation: the method of adjusting the proportion of the original image, which can be divided into two methods: reduction and enlargement.

5) Grayscale transformation: The method of changing the grayscale value of pixels in the original image to generate a new image. In this process, it is necessary to perform calculations according to the selected transformation function.

6) Noise disturbance: A method of artificially adding noise to the original image to generate a new image. Commonly used noises include salt and pepper noise and gaussian noise.

2.3. **Transfer Learning.** Transfer learning is the most cutting-edge research field in machine learning. Generally, it uses learned knowledge to learn new knowledge. Transfer learning is mainly used to search for labeled data from related fields for training when there is less labeled data in the target field. Its main goal is to quickly transfer the learned knowledge to a new field to complete or improve the learning effect of target fields or tasks [23, 24]. According to tasks, domains, and data availability, transfer learning can be roughly divided into three categories: inductive transfer learning, unsupervised transfer learning and direct transfer learning. Presently, almost all CNNs need to be fully trained on large image data sets to learn a large number of features and weights required for image classification and recognition. The pre-trained model used in this paper learns a lot of knowledge in the Imagenet data set, and then uses the model migration method to migrate and learn the data set in this paper. It saves a lot of time and makes the network converge quickly, which is helpful to improve the training speed and the performance of the model in the wild vegetable species recognition task.

3. Methods.

3.1. **Data set construction.** In order to improve the reliability and robustness of the recognition model and algorithm, the data set in this paper is mainly obtained in two ways. One is to use crawler technology to crawl the required image data on the relevant plant image database website and Baidu search engine, and save it in the corresponding folder. The second is to obtain relevant data through field shooting by smartphone. In order to avoid changes in image quality, the default setting of the phone is used, with a focal length of 4mm, an exposure time of 1/60 s and a resolution of 2280x1080.

Take the crawling process of the China Plant Image Library website as an example, the specific process is as follows:

1) Analyze web page structure

According to the characteristics of the Chinese Plant Image Database, each image leaf surface corresponds to a fixed id value, and the id belongs to a range of numbers. All image pages can be obtained by traversing this range. The display and operation of HTML page is based on the DOM tree, and the specific image elements and image meta information content in the image page can be analyzed.

2) Use Selenium tools for image acquisition

Selenium is a tool set for web automation testing. Selenium runs automatically in the browser and simulates the manual operation of real users. Therefore, the corresponding asynchronous loading content can also be obtained for web pages that use JS for asynchronous loading. Utilize this feature to simulate a user entering anurl composed of different ids in the search box, and then get the result picture of the page.

3) Image deduplication

Since images are crawled from multiple sources based on the vocabulary, there will be repeated identical images. In order to solve this problem, after the final image acquisition is completed, the image data set is de-duplicated. Based on the above two methods, total 3355 images of 10 wild vegetable plants were obtained. Divide the data set into three parts: training set, validation set, and test set at a ratio of 8:1:1. Among them, there are 2684 images in the training set, 336 images in the validation set and 335 images in the test set, respectively. The training set and the verification set are used for model training and verification in the training process by cross-validation, and the test set is used for model evaluation.The data is shown in Figure 2.

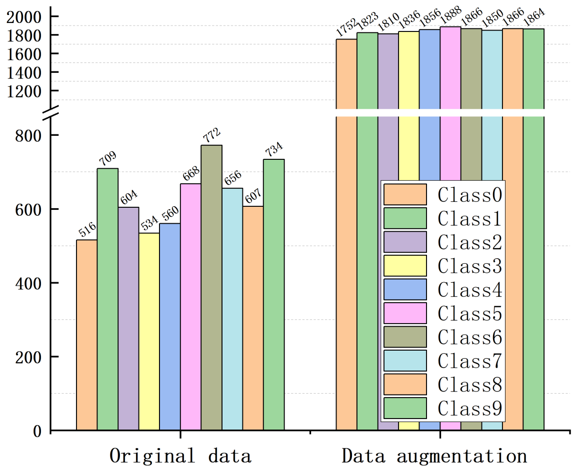


FIGURE 2. The wild vegetables dataset

3.2. **Data preprocessing.** In order to improve the accuracy and generalization ability of the model, the training data set is expanded before the model training. It mainly includes: random cropping of pictures, horizontal flipping of pictures, picture standardization, contrast enhancement, and brightness enhancement operations. The experimental data in this paper increase the brightness by 50% and the contrast by 30%. After this operation, the images in the training data set are expanded to 16,647, which is 6 times the original number.

Among them, the image random clipping operation is to randomly normalize the image of the training set to a value in [256, 512], intercept the upper left, upper right, bottom left, bottom right and center, and intercept the image into 224 pixels to increase the translation invariance. The image visualization results after data clipping preprocessing are shown in Figure 3.

Figure 3. The example of image clipping

In this paper, the adjustment of image color and contrast is completed by increasing the brightness and contrast. The brightness and contrast are increased by 60% and 30%, respectively. Then the image is flipped horizontally to increase the rotation invariance of the model. The effect after image preprocessing is shown in Figure 4 and Figure 5.

(a) Original (b) Brightness increase (c) Constrast increase

Figure 4. The effect of color dithering

Secondly, in order to improve the prediction accuracy of the model and accelerate the convergence speed, the mean value of the pixel points at the corresponding position of the training set picture is calculated. Normalize the picture using the mean value:

$$p_{(}x, y, z)' = p_{(}x, y, z), Z \in r, g, b \tag{2}$$

(a) Original

(b) Horizontal flipping

FIGURE 5. The effect of data flipping horizontal

The normalized value obtained by subtracting the mean pixel from the pixel value at the position is used for model training.

3.3. **Model training.** The classification model in this paper uses a CNN as a feature extraction network. The extracted features are passed through a fully connected layer and then the soft-max classifies the input image data. The data sets are placed in different folders by category, and the folders are named C0-C9. The wild vegetable image data set is divided into three parts: training set, verification set and test set according to the ratio of 8:1:1. The training set, verification set and test set are used for model training, model optimization, and to finally test the accuracy of model classification, respectively. The specific model training process is as follows (Figure 6):

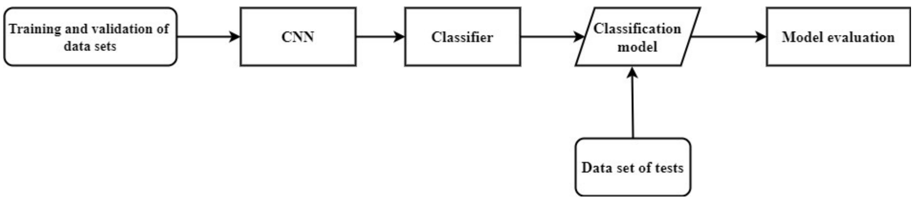


FIGURE 6. Training flow chart of models

4. **Experiment and result analysis.**

4.1. **Experimental environment.** The experimental environment of this article is Windows 10 64-bit system, using Tensorflow deep learning open source framework, and Python as the programming language. The hardware environment: the computer memory is 32Gb with Intel®Xeon(R) CPU@2.20 GHz processer, and GPU is NVIDIA GeForce GTX 1080 Ti with 11 G memory. Soft environment: CUDA Toolkit 9.0, CUDNN V7.0; Python3.6, Tensorflow-GPU 1.8.0.

4.2. **Comparative experiment.** This article mainly uses MobileNetV1, MobileNetV2, InceptionV3, VGG-16, and GoogleNet deep learning network to perform migration training on the training data set. According to the test results of the training model on the test data set, the parameter amount of the model, the memory size of the model, and a model suitable for transplantation to the mobile phone is comprehensively selected as a mobile phone recognition App software model. The hyperparameter settings in the model training are as follows: The initial learning rate is 0.01, and the training data set is divided into multiple batches by batch training. Each batch is trained with 537 images, the number of iterations per round is 31, and the epoch is set to 100.
Figure 7 shows how the recognition and classification accuracy of five networks on the training data set varies with the number of training rounds. When the number of training rounds reaches 50 rounds, the classification accuracy of the model reaches the best and tends to be stable. The MobileNetV2 network has a higher overall accuracy rate and faster convergence rate during the training process, and the MobileNetV1 network training accuracy rate and convergence rate are poor. The performance of VGG-16 and GoogleNet networks in the data set training process is basically the same.

The test results are shown in Table 1. It can be seen that the average prediction accuracy (AA) of the MobileNetV2 and VGG-16 networks in this data set is 96.92% and 95.31%, respectively, and the prediction results of the classification models after the training of the two networks are higher. Among them, the recognition accuracy (ACC) of MobileNetV2 to C2 category reached 99.05%, and that of VGG-16 to C3 category reached 98.86%. The recognition rate of 10 wild vegetable categories has reached more than 90%. Therefore, it is believed that these two recognition networks have played a better role in feature extraction in the classification of a single wild vegetable, and the classification model can achieve better results.

By analyzing the parameters and model size after model training, the classification model based on MobileNet network has certain advantages in recognition rate and memory ratio. This is because it adopts deep separable convolution operation in the feature extraction stage, and the parameters are 10% of the classification

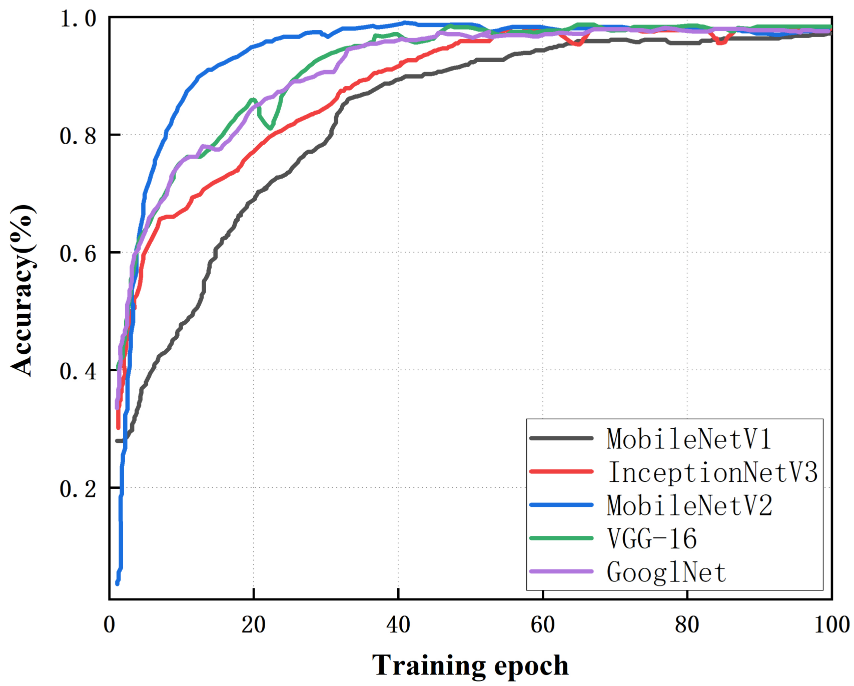


FIGURE 7. Prediction accuracy of five models on training data sets

TABLE 1. Classification results of testing sets

Category	Accuracy (ACC, %)				
	Mobile NetV1	**MobileNetV2**	**GoogleNet**	**VGG-16**	**InceptionV3**
C0	92.86	95.89	94.62	96.32	91.66
C1	93.65	96.63	93.86	92.81	91.87
C2	89.63	**99.05**	95.28	94.53	92.69
C3	92.58	97.38	92.68	**98.86**	96.48
C4	93.09	96.74	89.42	93.68	94.36
C5	94.23	92.68	94.05	95.92	89.04
C6	90.64	98.44	98.63	93.46	91.83
C7	85.72	96.38	90.55	95.78	96.58
C8	96.37	97.48	96.15	96.75	97.62
C9	92.92	98.65	93.69	94.98	94.48
Average accuracy (AA, %)	92.17	**96.92**	93.89	**95.31**	93.66

model based on VGG-16. After calculation, the final size of the five classification models after training is divided into 27.7Mb, 16.4Mb, 83.3Mb, 90.2Mb and 56.1Mb, respectively. The classification model based on MobileNetV2 has greater advantages in size than other models, which provides feasibility for transplantation to mobile terminal models.
Comprehensively analyze the prediction accuracy and memory proportion of the five models, this paper selects the MobileNetV2 based classification model as the mobile terminal wild vegetable recognition and classification App model. In order to test the predicted running time on the mobile phone, this paper migrates the trained model to the mobile phone. Through software detection, the running memory of the model is 28.89Mb, and the average running time is 418ms.

5. **Design and development of wild vegetable identification App on mobile terminal.**

5.1. **Development environment of Android mobile phone.** The development environment of this paper is Windows10(64 bit) operating system. The environment of Android system mainly includes java development kit (JDK), Android Studio software and Android software development kit (SDK). JDK is the core of the whole JAVA development, including JAVA running environment (JVM+JAVA system class library) and JAVA tools. Android studio is an integrated development environment (IDE) for Android application development. Android SDK is a free Android Software development kit provided by Google for developers. The development kit includes four parts: development code examples, interface files (APIs), development documents and development tools.

5.2. **Functional analysis and design of wild vegetable recognition App.** The core functions of the wild vegetable identification App designed in this paper mainly include: user management, image acquisition and image recognition functions.
1) User management function
Through this function module, users can realize registration, login and information modification, and use remote database to access and share resources. Users first need to register an account with an email. After registration, they can use the open function.
2) Picture acquisition function
The image acquisition function of the app is mainly realized in two ways: taking photos and selecting pictures from albums.
Firstly, it is necessary to judge whether there is wild vegetable picture to be identified. If no, call the camera of the mobile phone to take pictures, and load the taken photos into the system. If yes, load the picture directly into the system.

To take pictures by the camera of the Android system mobile phone, first turn on the camera of the system, and then check whether the camera can work normally. If it is normal, return to the previous step. If not, call the camera to take pictures. After the photos are taken, check whether the SD card exists and whether the remaining space of the SD card is sufficient. If there is enough space, save the photos on the SD card. The specific process is shown in Figure 8.

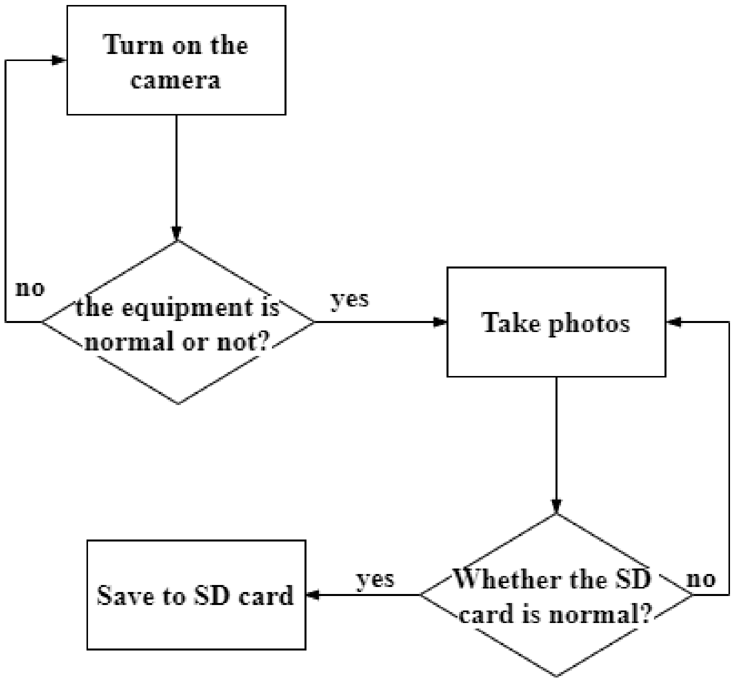


Figure 8. Flow chart of taking photos

Select a picture from an album. First, open the system album and check if the SD card is available. If the SD card is not available, exit the album selection. If available, read the media database and the image according to the address of the obtained image. The specific process is shown in Figure 9:

3) Picture recognition function

This function mainly displays the names of the first three categories with the highest classification probability through the identification and analysis of the uploaded pictures. Usually, the result with the highest classification probability is taken as the final classification result. This method can provide a choice for users to judge the model classification result according to the actual description.

In the picture recognition module, this paper adds the description information of the identified wild vegetable species. This information is realized by calling the description data in the database. The user compares and analyzes the collected data with the displayed data, judges the probability ranking of the comprehensive

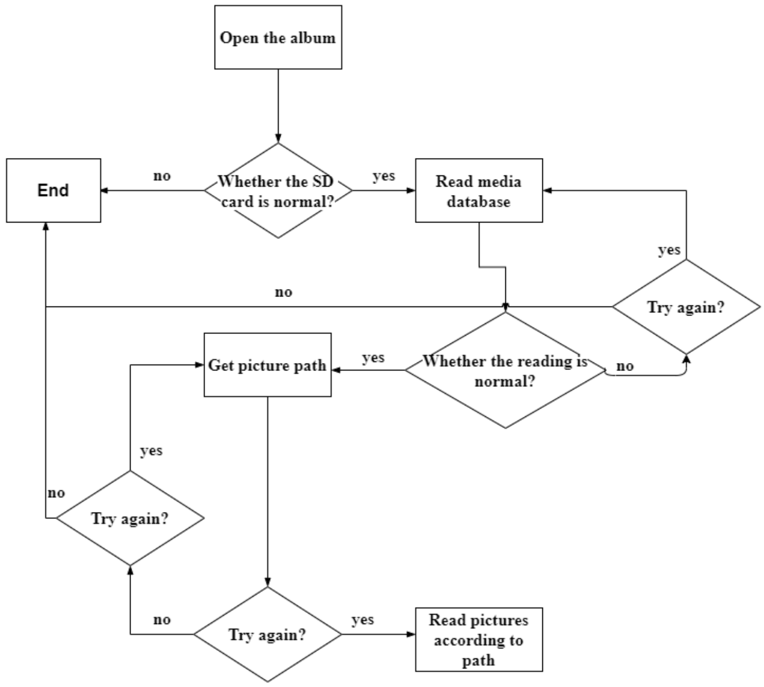


FIGURE 9. Flow chart of album selection module

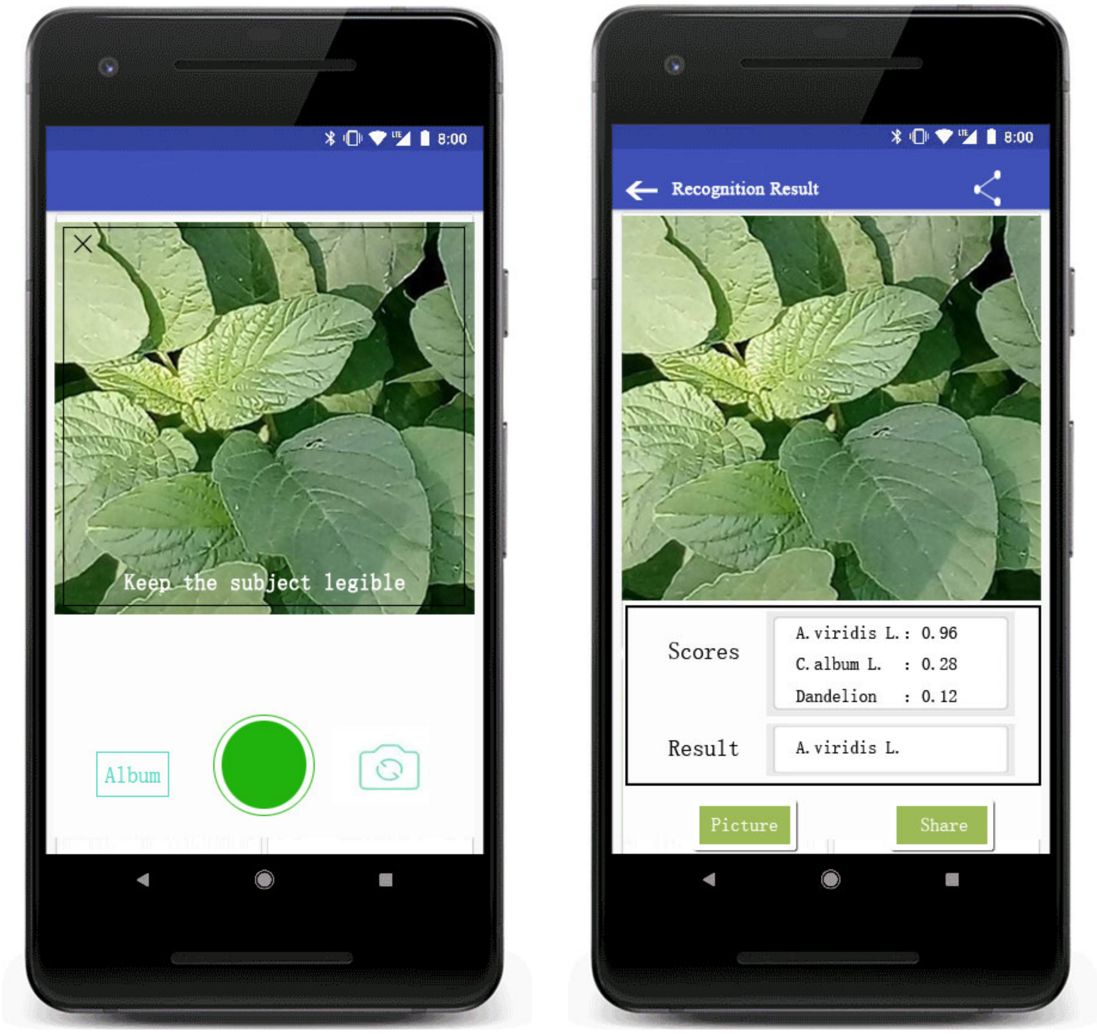
model, and finally confirms the wild vegetable species to be identified. If the discrimination results and description information are inconsistent with the actual data, the description information can be modified, and the results will be saved in the corresponding description information in the database to improve the recognition accuracy of this kind of wild vegetables.

5.3. **Implementation and test of wild vegetable identification App.** Based on the above analysis results, this paper uses android studio integrated development environment to develop the app. The specific process is: 1) model preparation; 2) Software installation and configuration; 3) Model import and parameter modification; 4) Testing and verification. Model preparation is mainly to obtain the file named by .pb and .txt, which are generated after computer training. Software installation and configuration mainly refers to the installation of development tools and corresponding SDK. In view of the model import and parameter modification, it needs to open the Project view in Android Studio, put .pb file and .txt file under app/src/main/assets, define the variables and initialize the Tensorflow interface, and configurate the version of (build gradle) corresponding to the file. Add 3 label buttons on the App page, namely album, camera, and recognition image. Then configure the button to call the TF model and execute "Build APK(S)" to generate an installable file package (APK file). Then, it can be transferred to the Android phone via 4G wireless network, or the installation package can be moved to the Android phone via a wired method, to complete the transplantation of the plant disease leaf app based on Android. The test mobile phone of the app is oppo (r15). Click to install the app on the android mobile phone. After the installation, select the wild vegetable image for testing and verification.

An example of mobile phone recognition is shown in Figure 10. Among them, shows the recognition homepage interface. The user can click the middle camera button to open the phone camera and choose to take a picture, or click the album button to select a photo in the local memory for recognition. The recognition result is shown in Figure 10(b). Three types with the highest classification probability scores are displayed in the lower part of the photo, and the result with the highest classification probability is used as the final recognition result. The result shown in Figure 10 is wild amaranth. In addition, users can also generate beautiful pictures and share the results of the recognition results, providing users with an autonomous experience.

6. **Conclusions.** In view of the low recognition efficiency of wild vegetable recognition and classification relying on traditional manual experience, based on image processing and deep learning technology, five CNN models are used to train the wild vegetable data sets constructed by us. The five classification models are compared and analyzed in terms of recognition accuracy, model size, and operation speed. We concluded that the recognition accuracy of the five CNN models on the wild vegetable data sets has obtained good results. ACC of MobileNetV2 to C2 category and VGG-16 to C3 category reaches 99.05% and 98.86%, respectively. These two recognition networks play a good role in feature extraction in single wild vegetable classification, and the classification model can achieve good results. And Comprehensive analysis of the prediction accuracy and memory ratio of the five models, this paper selects the MobileNetV2 based classification model as the vegetable recognition and classification app model on the mobile phone. After testing, this article transfers the trained model to the mobile phone. The running memory of the model is 28.89Mb, and the average running time is 418ms.

Above all, the wild vegetable recognition and classification App developed based on the model has better effect. Although the function is simple and easy, the recognition accuracy of common wild vegetable varieties is high. It can quickly and conveniently guide people to pick and use wild vegetable resources in practical applications. The app in this article has certain limitations. In the follow-up research, the focus will be on how to improve the existing CNN structure to achieve a more lightweight feature extraction network model, expand the data set category and increase the accuracy of the recognition app.



(a) Identification interface (b) Recognition result

FIGURE 10. The test effect of wild vegetables recognition application

Acknowledgment. This work was supported by the National Key R&D Program of the Ministry of Science and Technology- Research on Field Planting Online Monitoring Technology and System Standards(2018YFF0213602). We would like to express my gratitude to all those helped me during the writing of this paper.

REFERENCES.

[1] M.Y. Li, Z.G. Su, G.M. Yan, Present situation of exploitation-utilization and the developmental perspective of wild vegetable resources in China. *Heilongjiang Agricultural Sciences*, vol. 8, pp. 136-138, 2011.

[2] Aotegenbaiyin, Y.Q. Li, Y.H. Han, Z.M. Yan, X.J. Yang, T.Y Ma, The status of development and utilization of wild vegetables in China. *Journal of Hebei Agricultural Sciences*, vol. 19, no. 6, pp. 92-96, 2015.

[3] F. Zhou, *Resource investigation to wild vegetables in Hanzhong district and analyse and evaluate for nutritional compositions of characteristic species*, Ph.D. Thesis, Shaanxi Normal University,Shaanxi, China, 2004.

[4] G.H. Yue, Y.Z. Wang, A.H. Yu, Quicker reference atlas for field identification of 500 wild vegetables. *Letters in Biotechnology*, vol. 27, no.5, pp.692, 2015.

[5] J.R. He, Y.X. Shi, B. Liu, D.J. He, External quality grading method of Fuji apple based on deep learning. *Transactions of the Chinese Society of Agricultural Machinery.* 2021, http://kns.cnki.net/kcms/detail/11.1964.S.20210425.1319.014.html

[6] J. Y. An, *Research on wheat drought recognition and classification based on image depth learning*, Ph.D. Thesis, Beijing Chinese Academy of Agricultural Sciences, Beijing, China, 2019.

[7] L.R. Wei, J. Yue, Z.B. Li, G.J. Kou, H.P. Qu, Multi classification detection method of plant leaf diseases based on kernel function support vector machine. *Transactions of The Chinese Society of Agricultural Machinery*, vol.48, no. S1, pp. 166-171, 2017.

[8] J.M. Wu, Z.Li, N. Herencsar, B. Vo, J.C.W. Lin, A graph-based CNN-LSTM stock price prediction algorithm with leading indicators. *Multimedia Systems*, pp. 1-20, 2021.

[9] K. Tseng, J. Lin, C. Chen, M.M. Hassan. A fast instance segmentation with one-stage multi-task deep neural network for autonomous driving. *Computers & Electrical Engineering* , vol. 93, pp.107194, 2021.

[10] X.Q. Zhang, J.H. Chen, J.J. Zhuge, L.J. Yu, Deep learning based fast plant image recognition. *Journal of East China University of Science and Technology (Natural Science Edition)*, vol.44, no.6, pp.887-895, 2018.

[11] J.H. Zhang, F.T. Kong, J.Z. Wu, Z.F. Zhai, S.Q. Han, S.S. Cao, Cotton disease identification model based on improved VGG convolution neural network, *Journal of China Agricultural University*, vol.23, no.11, pp.161-171, 2018.

[12] T.T.N. Nguyen, T.L. Le, H. Vu, V.S. Hoang, Towards an automatic plant identification system without dedicated dataset, *International Journal of Machine Learning and Computing*, vol. 9, no. 1, 2019.

[13] P. Barré, B.C. Stöver, K.F. Müller, V. Steinhage, LeafNet: a computer vision system for automatic plant species identification, *Ecological informatics*, vol. 40, pp. 50-56, 2017.

[14] A. Elhassouny, F. Smarandache, Smart mobile application to recognize tomato leaf diseases using convolutional neural networks, *2019 International Conference of Computer Science and Renewable Energies (ICCSRE)*, 2019.

[15] M. Dyrmann, H. Karstoft, H.S. Midtiby, Plant species classification using deep convolutional neural network, *Biosystems Engineering*, vol. 151, pp. 72-80, 2016.

[16] A.G. Howard, M.L. Zhu, B. Chen, D. Kalenichenko, W.J. Wang, T. Weyand, M. Andreetto, H. Adam, MobileNets: efficient convolutional neural networks for mobile vision applications, *arXiv preprint*, arXiv:1704.04861, 2017.

[17] Y. Liu, Q. Feng, S.Z. Wang, Plant disease identification method and mobile terminal application based on lightweight CNN, *Transactions of the Chinese society of agricultural engineering*, vol. 35, no. 17, pp. 194-204.

[18] L. Gang, H.X. Zhang, E. Linning, L. Yu, Z. Juming, Recognition of honeycomb lung in CT images based on improved MobileNet model, *Medical physics*, vol. 48, no. 8, pp. 4304-4315, 2021.

[19] S. Prasad, S.K. Peddoju, D. Ghosh, Multi-resolution mobile vision system for plant leaf disease diagnosis. *Signal, Image and Video Processing*, vol. 10, no. 2, pp. 379-388, 2016.

[20] M.N. Qiu,*The study on small sample sets facerecognition based on sample augment method*, Ph.D. Thesis, Harbin Institute of Technology, Harbin, China, 2014.

[21] A. Gupta, A. Vedaldi, A. Zisserman, Synthetic data for text localisation in natural images, *Proceedings of the IEEE conference on computer vision and pattern recognition*. pp. 2315-2324, 2016.

[22] X.R. Yu, Remote sensing image analysis based on data amplification depth learning, *Shandong: China University of Petroluem*, MA.Eng. Thesis, Shandong, China, 2018.

[23] X.P. Liu, X.D. Luan, Y.X. Xie, M.Z. Huang, Transfer Learning Research and Algorithm Review, *Journal of Changsha University*, vol. 32, no. 5, pp. 28-31+36.

[24] L.C. Zhang, Research on application development based on Android platform architecture, *Computer programming skills and maintenance*, vol. 6, pp. 52-53+85, 2017