2023 4th International Symposium on Computer Engineering and Intelligent Communications (ISCEIC) | 979-8-3503-0632-3/23/$31.00 ©2023 IEEE | DOI: 10.1109/ISCEIC59030.2023.10271209

Transfer Learning-Based Flower Image Classification: Leveraging The Pre-Trained Alexnet Model

Ruitong Xiao*
College of Computer and Network Security (Oxford Brookes University)
Chengdu University of Technology
Chengdu, China
13508017709@163.com

Rui Wang
Engineering College
Chengdu College of University of Electronic Science and Technology of China
Chengdu, China
3108616557@qq.com

***Abstract*—Image classification performs an influential role in the field of computer vision, while deep learning has brought remarkable progress for image classification. Targeting the flower image classification task, this paper proposes a transfer learning-based method to achieve effective flower classification using a pre-trained AlexNet model. This method chooses AlexNet as the base pre-trained model and adapts it to optimize flower classification. To enhance the generalization capability of the model, the training images were resized and data augmented, and appropriate training options were defined to evaluate the model's classification accuracy through training. Through the analysis of experimental results, this method validates the effectiveness of the proposed method and results in a good performance on the flower image classification, which offers important implications for classification applications and provides strong support for botanical research, flower appreciation, and horticulture, among others.**

Keywords-component; Image classification, Transfer learning, AlexNet, Flower classification

I. Introduction

Image classification is the process of classifying images into different classes or categories based on the visual content of the image, which has a wide prospect of application in the field of computer vision that is applied to various fields such as target recognition, scenario comprehension, and medical image analysis [1]. However, there are specific problems and challenges faced in the flower image classification. Flower image classification involves accurate identification and classification of flower appearance, morphological features, and color. Nevertheless, the variety diversity of flowers and the complexity of flower shapes make the classification task challenging. Different flower varieties may have similar appearance features, while flowers of the same variety may differ in shape and color [2], [4]. In addition, flower image classification faces the influence of factors such as variation in lighting conditions and variation in shooting angles [9]. Moreover, due to the difficulty of acquiring and labeling flower image data, it suffers from the lack of large-scale and diverse datasets [3]. These factors increase the complexity and challenge of floral image classification.

In the field of image classification, traditional image classification methods generally rely on hand-designed feature extraction and classifiers, which require large amounts of manual involvement and field knowledge, as well as potentially exhibit limitations when dealing with complex floral images [8]. Hand-designed feature representations may fail to capture the fine-grained and abstract features in floral images, leading to degradation in classification accuracy [2], [3]. Additionally, traditional methods may confront efficiency and scalability issues when dealing with large-scale datasets and diverse floral images [1].

In recent years, the rapid development of deep learning has revolutionized the field of image classification. In particular, transfer learning has exhibited great potential as an effective method for floral image classification tasks [10]. Transfer learning, which involves employing pre-trained models on large datasets (e.g., ImageNet) and fine-tuning them on smaller datasets for specific tasks (e.g., floral image classification), it is possible to reduce the duration and computational resources required to train models from scratch and remarkably improve the generalization capability of the models [5], [6].

II. Aim

This paper presents a transfer learning-based method to solve the flower image classification problem. AlexNet model is a deep convolutional neural network architecture [7], this method selects AlexNet model as the pre-training model, based on which the last three layers of AlexNet are modified and adjusted to the output layer proper for flower classification. Furthermore, preprocessing of the training data, including image resizing and data augmentation, is performed to improve the robustness and generalization of the model. A dataset containing several flower species, including daisies, dandelions, roses, sunflowers, and tulips, etc., was practically implemented for this method and divided into a training set and a validation set. By training the model on the training set and evaluating the performance of the proposed method on the floral image classification tasks on the validation set, the predicted labels of selected images are displayed.

III. Methodology

This section aims to implement transfer learning-based flower image classification tasks. Considering the importance and wide application of image classification in the field of computer vision, especially the problems and challenges faced in floral image classification tasks, the development of techniques such as deep learning and transfer learning offer great possibilities for improving classification accuracy and generalization.

A. Flower Image Dataset

In the case of using transfer learning, datasets with dozens or hundreds of images are sufficient to train an effective Convolutional Neural Network (CNN) model [11]. A dataset containing 3500 images of various flowers that covers 5 flower categories with approximately 700 images in each category. The dataset was divided into a training set and a test set, where 2625 images (about 75% of the total) are utilized to train the model and the remaining 875 images (about 25% of the total) are utilized to test the performance and generalization capability of the model. This large-scale dataset provides sufficient samples and a diversity of classification categories to train and evaluate the floral image classification model, the specific flower image samples are shown in Fig. 1.

Figure 1. Flower Image Samples

To ensure the correctness and consistency of the data, this experiment applies data preprocessing to the flower image dataset. First, load the pre-trained AlexNet model as the base model. After that, using the imageDatastore object to load the flower image dataset and associate images with category labels through parameter settings to determine that the images are associated with the correct categories and stored.

B. AlexNet Convolution Neural Network (AlexNet CNN)

AlexNet is a model consisting of five convolutional layers and three fully-connected layers, where the last fully-connected layer is connected to 1,000 categories and the rest acts as a feature extractor. To adapt the AlexNet model to the flower image classification task (5 categories) in this paper, it was fine-tuned. During the fine-tuning process, the first few layers of the pre-trained AlexNet model were retained as fixed feature extractors, while the last three layers were replaced in order to adapt the model to the new task, i.e., classifying 5 categories.

During the fine-tuning process, a fully-connected layer with 64x64 filter size was added to accommodate the new output (5 categories), which became the 23rd layer of the fine-tuned AlexNet. In order to enhance the nonlinear problem-solving capability of the network, a Rectified Linear Unit (ReLU) layer (Layer 24) was also added, which provides nonlinear activation and speeds up the training process while avoiding the problem of gradient vanishing.

Furthermore, a fully connected layer (layer 25) with five output neurons was added in order to achieve classification of the five floral categories, while the weight of the last fully connected layer was initialized to 10, and the neuron biases in this layer were initialized to 20 using the constant 20 in order to allow the new layer to learn faster with respect to the transfer layer, speeding up the early learning process of the network. With the above modifications, the pre-trained AlexNet model is fine-tuned into a new model suitable for the flower image classification task.

In order to improve the generalization capability and resistance to overfitting of the model, this experiment performed data augmentation operations on the training images, using the imageDataAugmenter function to define the data augmentation, including a random flip in the horizontal direction and random panning of the images in the horizontal and vertical directions. The training dataset is then processed using the augmentedImageDatastore function to automatically resize the images and apply the data augmentation to the training images.

C. Training Options

The experiment conducted 12 training tests that covered three different types of training tests: mini batch size test, max epochs test, and initial learn rate test. each training test type contained four training tests aimed at discovering the best parameter settings to improve model performance and accuracy.

1) Stochastic Gradient Descent with Momentum (SGDM): This experiment utilizes the Stochastic Gradient Descent with Momentum (SGDM) as an optimizer, which is a commonly used optimization algorithm and can accelerate the training process as well as improves the convergence of the model and effectively optimize the model parameters.

2) Mini Batch Size: It is the batch size of images used for training in each iteration cycle. This experiment will determine the optimal mimi batch vsize by trying them with values of 32 64, 128 and 256.

3) Max Epochs: It is the maximum number of iterations for training. In different training, this experiment will determine the optimal max epochs value by trying max epochs with values of 4, 8, 10 and 16.

4) Initial Learn Rate: This is the learning rate used to update the model parameters at the beginning of training. In this experiment, initial learning rates of 0.001, 0.01, 0.0001, and 0.00001 were tested to validate the optimal initial learning rate.

5) Shuffle: Indicates whether the training data is randomly shuffled in each iteration cycle. It was set to "every-epoch" in all tests, i.e., randomly scrambling was performed in every cycle.

6) Validation Data: The validation data set used to validate the model performance.

7) Validation Frequency: Specifies how many iterations to perform a model validation. It was set to 30 iterations in all tests.

8) Verbose: Controls whether or not detailed information about the training process is displayed in the command window. It is set to "true" in all the tests, that is, to display detailed information.

9) Plots: Controls whether or not the training process is plotted. Set to "training-progress" for all tests, i.e. to plot the training progress.

10) Validation Accuracy: Validation accuracy results for each group of training options.

11) Validation Loss: Validation loss values for each training option.

12) Maximum Iterations: The maximum number of iterations for each training option.

13) Elapsed Time: Time spent on training for each training option.

14) Predicted Labels: Classification results obtained after the model predicts the input samples.

IV. RESULT

The experiment conducted 12 training tests, each training test type contained four training tests, and the following will show the logging tables for each test, as well as the analysis and summary of the results.

A. Mini Batch Size Test Results

TABLE I. TEST RESULTS FOR MODIFYING MINI BATCH SIZE

Training Option	Value			
Solver Name	sgdm	sgdm	sgdm	sgdm
Mini Batch Size	32	64	128	256
Max Epochs	4	4	4	4
Initial Learn Rate	0.001	0.001	0.001	0.001
Shuffle	every-epoch	every-epoch	every-epoch	every-epoch
Validation Data	augimdsValidation	augimdsValidation	augimdsValidation	augimdsValidation
Validation Frequency	30	30	30	30
Verbose	true	true	true	true
Plots	training-progress	training-progress	training-progress	training-progress
Validation Accuracy	29.52%	82.00%	85.81%	84.10%
Validation Loss	0.6745	0.4953	0.4218	0.4748
Maximum Iterations	304	152	76	36
Elapsed Time	17min6sec	17min0sec	11min13sec	15min35sec

B. Max Epochs Test Results

TABLE II. TEST RESULTS FOR MODIFYING MAX EPOCHS

Training Option	Value			
Solver Name	sgdm	sgdm	sgdm	sgdm
Mini Batch Size	128	128	128	128
Max Epochs	4	8	10	16
Initial Learn Rate	0.001	0.001	0.001	0.001
Shuffle	every-epoch	every-epoch	every-epoch	every-epoch
Validation Data	augimdsValidation	augimdsValidation	augimdsValidation	augimdsValidation
Validation Frequency	30	30	30	30
Verbose	true	true	true	true
Plots	training-progress	training-progress	training-progress	training-progress
Validation Accuracy	85.81%	87.71%	88.38%	88.00%
Validation Loss	0.4218	0.4253	0.4218	0.3611
Maximum Iterations	76	152	190	304
Elapsed Time	11min13sec	26min0sec	24min27sec	118min48sec

C. Initial Learn Rate Test Results

TABLE III. TEST RESULTS FOR MODIFYING INITIAL LEARN RATE

Training Option	Value			
Solver Name	sgdm	sgdm	sgdm	sgdm
Mini Batch Size	128	128	128	128
Max Epochs	10	10	10	10
Initial Learn Rate	0.001	0.01	0.0001	0.00001
Shuffle	every-epoch	every-epoch	every-epoch	every-epoch
Validation Data	augimdsValidation	augimdsValidation	augimdsValidation	augimdsValidation
Validation Frequency	30	30	30	30
Verbose	true	true	true	true
Plots	training-progress	training-progress	training-progress	training-progress
Validation Accuracy	89.52%	20.00%	85.52%	84.19%
Mini-batch Accuracy	95.31%	19.53%	89.84%	75.00%
Validation Loss	0.3816	1.6106	0.4355	0.4808
Mini-batch Loss	0.0965	1.6117	0.2962	0.6953
Elapsed Time	19min56sec	26in05sec	27min53sec	29min22sec

By analyzing and summarizing the test results in Tab. 1, it can be found that when the mini batch size is 128, the validation loss is the lowest (0.4218) and the validation accuracy is the highest (85.81%), and the duration is the shortest (11min13sec). In Tab. 2, the mini batch size is fixed to 128. As the max epoch increases, the validation accuracy gradually increases while the validation loss gradually decreases, where the best max epoch is 10 and its validation accuracy is 88.38%. Subsequently, the mini batch size was fixed to 128 and the max epoch was fixed to 10 to explore the effect of the initial learning rate on the validation accuracy, etc. It was found in Tab. 3 that when the initial learn rate was set to 0.001, the best validation accuracy (89.52%) and lower validation loss (0.4218) were obtained. Whereas, at higher (0.01) or lower (0.0001 and 0.00001) initial learn rates, the performance of the model was obviously degraded, resulting in poorer performance in terms of validation accuracy and loss function.

After determining the best training options, the model was trained utilizing these options, as shown in Fig. 2 for the training progress plot, the horizontal axis indicates the number of training iterations and the vertical axis indicates the corresponding accuracy and loss values. As shown in Fig. 3, randomly selected flower images are successfully recognized and displayed with the corresponding labels.

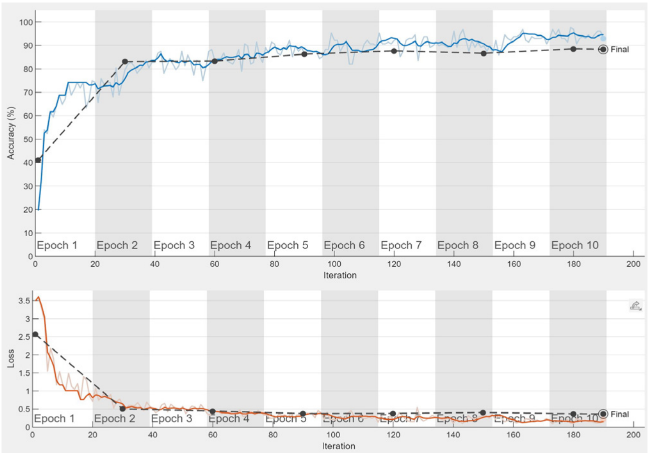

Figure 2. Training and validation results

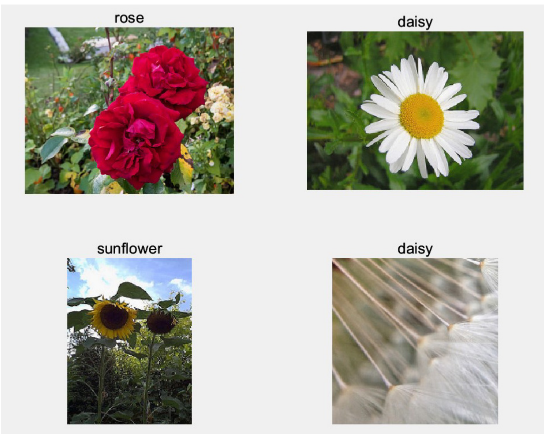

Figure 3. Flower recognition results

In summary, when the mini batch size is 128, the max epoch is 10 and the initial learn rate is 0.001, higher validation accuracy and lower loss as well as shorter elapsed time will be obtained in comparison. However, due to the fact that AlexNet has a large number of parameters and a complex network structure, it requires relatively high computational capabilities to process and train the model, perhaps requiring more memory and computational resources to optimize the training results.

V. Conclusions

This paper explores the application of a transfer learning-based method to the flower image classification tasks. Through employing the pre-trained AlexNet model as the base model, fine-tuning and training with flower recognition requirements, the accurate classification of flower images is successfully achieved. The results of this study are relevant for practical flower recognition and classification applications and provide useful support for the fields of botanical research, flower appreciation and horticulture. In further exploration, different pre-training models and data enhancement methods may be experimented to further improve the accuracy and robustness of flower image classification.

References

[1] E. Vocaturo, “Image Classification Techniques,” pp. 22–49, Jan. 2021, doi: https://doi.org/10.4018/978-1-7998-2742-9.ch003.

[2] J. Hu and Q. Li, “Flower image classification based on improved inception V4 network,” Oct. 2022, doi: https://doi.org/10.1117/12.2639679.

[3] Gao Yifei, Qiu Chuxian, Xu Jiexiang, Miao Yixuan, and Teoh Teik Toe, “Flower image classification based on improved convolutional neural network,” Nov. 2022, doi: https://doi.org/10.1109/itme56794.2022.00028.

[4] L. Lu, H. Ying, H. Mou, Z. Sun, and W. Qian, “Flower image classification based on visual transformer,” Oct. 2022, doi: https://doi.org/10.1117/12.2657133.

[5] J. Plested and T. Gedeon, “Deep transfer learning for image classification: a survey,” May 2022, doi: https://doi.org/10.48550/arxiv.2205.09904.

[6] G. Ayana, J. Park, J.-W. Jeong, and S. Choe, “A Novel Multistage Transfer Learning for Ultrasound Breast Cancer Image Classification,” *Diagnostics*, vol. 12, no. 1, p. 135, Jan. 2022, doi: https://doi.org/10.3390/diagnostics12010135.

[7] A. A. Almisreb, N. Jamil, and N. M. Din, “Utilizing AlexNet Deep Transfer Learning for Ear Recognition,” *2018 Fourth International Conference on Information Retrieval and Knowledge Management (CAMP)*, Mar. 2018, doi: https://doi.org/10.1109/infrkm.2018.8464769.

[8] M. Tian and Z. Liao, “Research on Flower Image Classification Method Based on YOLOv5,” *Journal of Physics: Conference Series*, vol. 2024, no. 1, p. 012022, Sep. 2021, doi: https://doi.org/10.1088/1742-6596/2024/1/012022.

[9] Rohan de Silva and Pavithra Senanayake, “Gender Identification of Imperfect Flowers Using Image Classification,” Oct. 2018, doi: https://doi.org/10.1109/atc.2018.8587564.

[10] R. Murugeswari, Kasi Vishwanath Nila, V. Dhananjeyan, B. Sai, and K. Prabhas, “Flower perception using Convolution Neural Networks based Escalation of Transfer learning,” Jan. 2022, doi: https://doi.org/10.1109/icssit53264.2022.9716338

[11] J. Shell and S. Coupland, “Fuzzy Transfer Learning: Methodology and application,” Information Sciences, vol. 293, pp. 59–79, Feb. 2015, doi: https://doi.org/10.1016/j.ins.2014.09.004.