Review reply

Reply to review of team 86:
Thank you for your review and your high evaluation.
(1)We did use VGG network as one of CNN networks to classify images.

Reply to review of team 65:
Thank you for your review and your high evaluation.
(1)We should add more famous neural network like resnet and desnet to compare results, but considering work load we did not do it.
(2)For datasets, we pre-processed our training set by image rotating,copped and etc. So we did image augmentation for datasets.

Reply to review of team 30:
Thank you for your review and your high evaluation.
(1) For datasets, we pre-processed our training set by image rotating,copped and etc. So we did image augmentation for datasets.
(2) We tried different learning rates in VGG model. After experiments, we found the best result could be achieved by learning rate 0.05. Maybe we should explore why this learning rate gave the best result.
(3) We should add more famous neural network like resnet and desnet to compare results, but considering work load we did not do it.

Flower Classification by SVM and CNN

JunFeng Zhao
Group5
juz016@eng.ucsd.edu

Linyan Zheng
Group5
l5zheng@eng.ucsd.edu

Yening Dong
Group5
y2dong@eng.ucsd.edu

Abstract

In this paper, we propose and compare SVM and CNN on flower classification problem, which is a interesting and meaningful task because of the intra-class variation, inter-class similarity and occlusion. We choose Oxford 102 category flower dataset, and apply SVM and a series of VGGs with fully-connected layers on it. In terms of CNN part, we use VGG networks to extract features and fully-connected layers with softmax to classify flowers. As a result, CNN with VGG16 achieves 94.4% accuracy Our approach, but SVM only reaches 19.2%. On the other hand, deeper layers and lower learning rate have a positive influence on CNN performance.

1. Introduction

There exists 369,000 named species of flowering plants all over the world. Therefore, it is difficult to distinguish these flowers for most ordinary people except experienced plant experts. We usually ask specialists about species of different flower plants when we want to know about what a kind of flowers is. Based on the development of modern electronics industry, an effective way to identify flower species can be done by identifying flower images, especially with the wide use of massive apps on smart phones.

Flower classification has a big meaning, however, many real restrictions limit its realization. Unlike other obvious objects which people are able to distinguish distinct categories in daily life such as houses or cars, flower classification is a much more challenging task because of inter-class similarity and inter-class variation. Even we can say that there does not exist two flowers which are exactly the same in the world. In addition, it is very difficult to distinguish the difference in some kinds that are similar in appearance even for experienced plant experts. Figure 1(a) shows examples of inter-class similarity, there are totally three different kinds of flowers but very similar in appearance. Figure 1(b) shows inter-class variability due to the difference in illumination and color. What is more, images of flowers are often taken in a real environment where the illumination condition varies with weathers and time. Also, there is a lot more variation in viewpoints, occlusions, the scale of flower images. And, the background also makes classification task difficult. All these problems lead to a confusion across classes and make the task of flower classification more challenging.

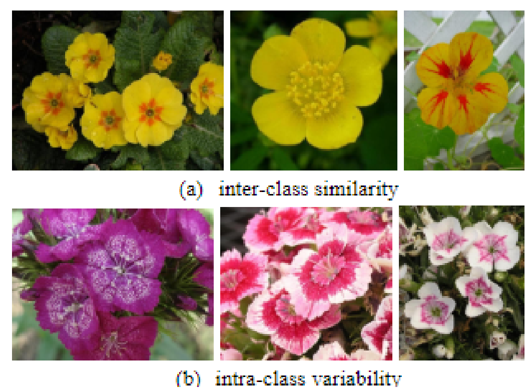

Figure 1. Diversity of Flowers

2. Related works

Flower Classification is a challenging and important problem in the botanical area, and there are a lot of people working on it. In [1] [2], Zisserman and Nils-back made a classification system by using visual vocabularies. In [3], Guru adopted a flower classification system by using KNN classifiers. In [4], SIFT features and feature contexts methods can encode different kinds of information like local and spatial, and classification part is solved by SVM. Kanan and Cottrell [5] designed a model using combination sequential visual attention using gaze and sparse coding. In [6], they focus on the fine-grained recognition, and they use a new detection and segmentation algorithm which applies zoom on objects to solve it. In [7], Savakar and Anami used a BPNN classifier based on color, features and textures to extract and recognize different agricultural plants. In [8], they use neural network to learn features and neural network and logistic regression to predict classes.

3. Dataset and Features

The Oxford University provides us with two flower datasets, including The Oxford 17 and The Oxford 102. The latter one is much larger, consisting of 102 different categories of flowers which are the most common in United Kingdom, so we choose this one as our dataset. It has 8189 images with 102 categories, and each of them have 40 258 images. The Figure 5 shows certain example of the dataset.

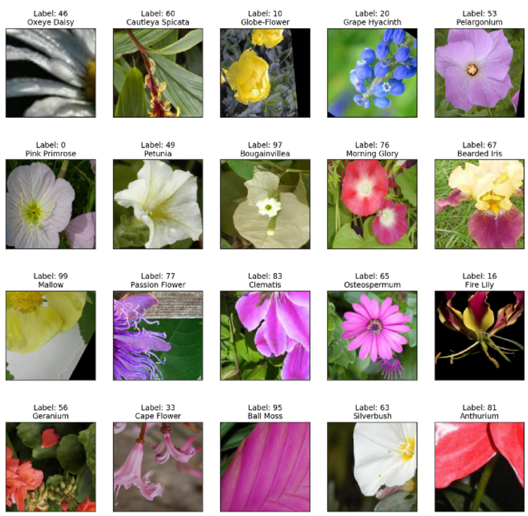

Figure 2. Data Sample in The Oxford 102.

For the data processing, we divided the dataset randomly into three sets, 70% for training set, 15% for validation set, and the other 15 % for testing set, because our model doesn't have many hyperparameters. This means our model is easy to validate, even using small size of dataset. After data splitting, we do some transformation like flipping and rotation to bigger variation in the dataset. In transformation, each sample chooses different methods randomly. Next we resize images to fixd sizes, then only crop the central part which are 224*224 for VGG and 128*128 for SVM, which can reduce surrounding's effect. We pick 224*224 for VGG, because the input size of VGG networks are required to be 224*224. On the other hand, we use a different image size for SVM. since we need to smaller image size to improve training efficiency, while input sizes won't make a difference on accuracy. Furthermore, we transfer image into tensor which are kinds of vector in Pytorch. Finally, we calculate mean and standard deviation to normalize tensor to increase consistency.

4. Methods

4.1. SVM

Support vector machine is a widely used machine learning method for classifying data.Basically, The objective of the support vector machine algorithm is to find a hyperplane in an N-dimensional space(N — the number of features) that distinctly classifies the data points.To separate the two classes of data points as shown in Figure 3. There are many possible hyper-planes that could be chosen. Our objective is to find a plane that has the maximum margin, i.e the maximum distance between data points of both classes. Maximizing the margin distance provides some reinforcement so that future data points can be classified with more confidence.In our machine learning model, we use HOG algorithm to extract features from input dataset which is better in feature extraction. Then we use LinearSVC()(SVM) to seperate features extracted by HOG.

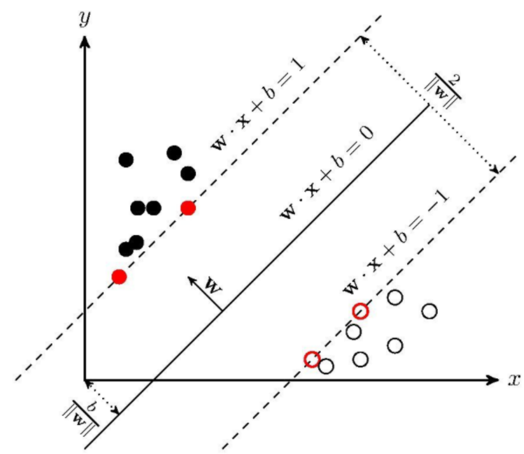

Figure 3. SVM classification

4.2. Convolutional Neural Network

Convolutional Neural Networks are a special kind of multi-layer neural networks. Like almost every other neural networks they are trained with a version of the back-propagation algorithm. Where they differ is in the architecture. Convolutional Neural Networks are designed to recognize visual patterns directly from pixel images with minimal preprocessing. They can recognize patterns with extreme variability, and with robustness to distortions and simple geometric transformations. In recent years, the CNN model has been widely applied in the field of image processing, so the image classification is to extract features from the dataset, and to classify them into a specific category.

The convolution neural network is one of the mostly used neural networks, which is composed of convolution layers and pool layers to extract features. The traditional convolution neural network structure is shown in Figure 4. The input layer is used to load the image which means processing the raw data for the neural network and the output vector is composed of multiple feature maps. The convolution layer is used to extract the target features by the convolution operation. These features are passed to the pool layer, which can reduce network parameters to speed up training process. At the end of CNN, it is ended by one or more fully connected layers(FC). The main effect of the fully connected layer is to process the extracted features into distinguishable class information. The last layer is the output layer, and the output of the neural network model is transformed into a probability distribution through the Softmax layer to obtain the probability information of the target category.

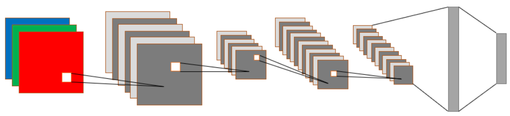

Figure 4. Traditional CNN Architecture

4.3. VGG Series

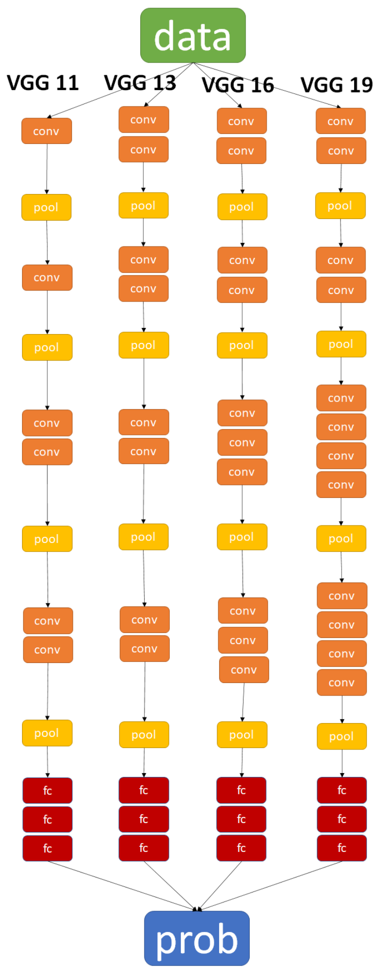

Figure 5. VGG Series Architecture

The VGG network was developed by the Visual Geometry Group team of Oxford University. The main purpose of the project is to prove that increasing the depth of the network can improve the accuracy of the network to a certain extent. The structures of VGG 11, 13, 16 and 19 are shown respectively in Figure 5, in which orange boxes represent convolutional layers, yellow boxes represent pooling layers and red boxes represent fully connected layers. The following is the more detailed network architecture and features. So more specifically, VGG-16 network has 16 layers which has parameters. The VGG-16 network structure is very regular and there are not so many hyper-parameters, focused on building a simple network. VGG-16 networks are several convolutional layers followed by a pooling layer that can compress the image size. All cvolutional layers use 3*3 small convolution kernels(use same padding) and 2*2 maximum pooling layers. According to the existing experimental results of VGG architectures, VGG-16 is considered to have the best performance.

5. Experiment

In this part, we show our model performance on The Oxford 102, there are three experiments, including SVM, different VGGs and different learning rates, all experiments are implemented on UCSD DataHub whose environment is 1 GPU, 8 CPU, 16G RAM.

5.1. Experimental Strategy

There are three experiments implemented to verify our hypothesis. At first, we use a traditional ML method SVM with HOG to classify flowers. Then we use three different VGGs, VGG11, VGG13 and VGG16, to name a few, to figure out the influence of network depth. Finally, we pick up the best feature extraction network, VGG16, to learn how learning rates will effect accuracy.

5.2. SVM Experiment

In order to get higher accuracy, we use HOG to extract features. We break 360 degrees into 12 bins, each 30 degrees, and use 8*8 pixels in per cell, and set 16 cells in each block, because the input size is 128*128. After feature extraction, we use Linear Support Vector Classification, it implements in terms of liblinear rather than libsvm, so it has more flexibility in the choice of penalties and loss function, and scale better to large numbers of samples. However, SVM doesn't work well, it only achieves 19.2%.

5.3. VGG Experiments

In this series experiments, we use the SGD optimizer with 0.001 learning rate, 64 mini-batch and momentum 0.9. In the following, we will do a specific learning rate experiment, so this time we just pick up a common default learning rate. In terms of mini-batch size, 32 is a good default, but it seems a little small for us. Because GPU resources on

UCSD Datahub is limited and kernel may crash, we have to consider time efficiency, 64 mini-batch can save certain time. Figure 6 and Figure 7 show the result of loss and accuracy curve for different VGG networks respectively, and Table 1 shows the final classification accuracy.

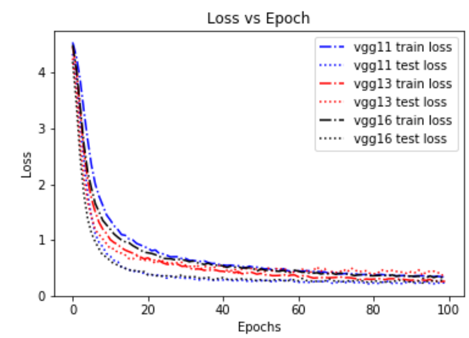

Figure 6. Loss Function for VGGs.

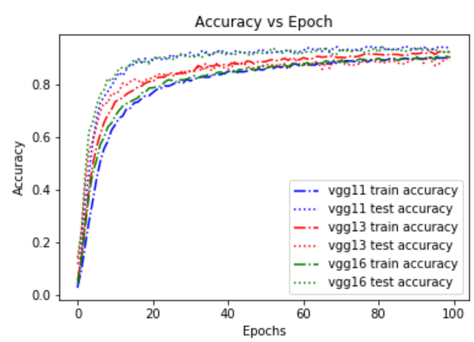

Figure 7. Accuracy for VGGs.

Table 1. Accuray for VGGs

Network	Learning Rate	Accuracy
VGG 11	0.001	93.9%
VGG 13	0.001	94.3%
VGG 16	0.001	94.4%
SVM	/	19.2%

From above figures, three VGG networks all converge after 50th epoch, and their curve tendencies are very similar, especially after convergence. Also, the final accuracy of them are quite close, around 94%. This means all three VGG networks work well on this flower classification, and the depth of networks doesn't have a big difference on this topic. The reason is that the stand deviation of samples in The Oxford 102 is small and we do lots of data processing, which removes noise in images, so it isn't that hard to classify these flowers. Furthermore, CNN is much better than SVM, because CNN extracts more features than HOG, and the classifier are more complicated.

5.3.1 Learning Rate Experiment

This part, we choose VGG 16, which has the best performance on this topic, with four different learning rates: 0.001, 0.005, 0.01 and 0.05. This range of learning rate covers most learning rates people used. The figure 8 shows the loss and accuracy plots, in terms of learning rates.

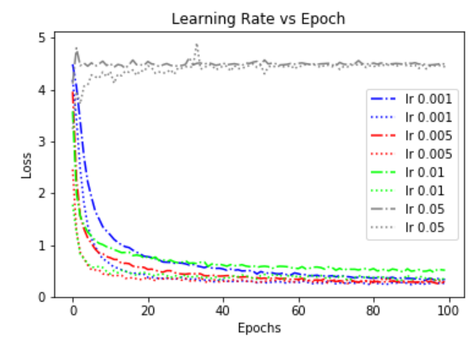

Figure 8. Loss of VGG with Different Learning Rates

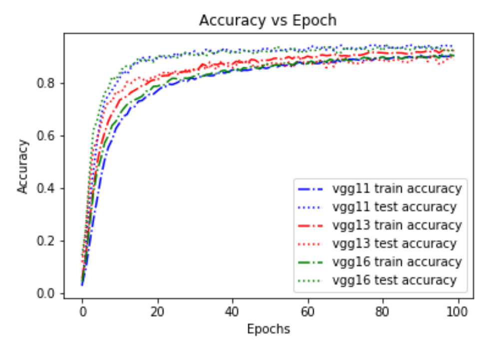

Figure 9. Accuracy of VGG with Different Learning Rates

Table 2. Accuray for VGG of Different Learning Rates

Network	Learning Rate	Accuracy
VGG 16	0.001	94.4%
VGG 16	0.005	92.8%
VGG 16	0.01	91.8%
VGG 16	0.05	11.2%

From Table 2, we can see that smaller learning rate has higher accuracy in our range, and 0.05 learning rate is so large that gradient descent cannot work and the training error increases. On the other hand, a good learning rate also should have faster convergence. Hence, 0.001 may not be the best one.

6. Conclusion

In this project, we propose SVM and CNN to solve flower classification problem, using The Oxford 102 folwer dataset. In all, CNN is much better than SVM, and VGG 16 with 0.001 learning rate achieves the highest accuracy 94.4%. Additionally, depth of networks doesn't have a big impact on this problem, but learning rate does. In the future, we will adapt a more complicated classifier to purse higher accuracy, and solve some occlusion problem.

7. Contribution

Junfeng is responsible for SVM, Linyan does learning rate experiments, and Yening built VGG series models. All of us works on proposal, presentation and report.

References

[1] M. Nilsback and A. Zisserman, "A Visual Vocabulary for Flower Classification," 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), New York, NY, USA, 2006, pp. 1447-1454.

[2] M. Nilsback and A. Zisserman, "Automated Flower Classification over a Large Number of Classes," 2008 Sixth Indian Conference on Computer Vision, Graphics Image Processing, Bhubaneswar, 2008, pp. 722-729.

[3] D.S.Guru, Y.H.Sharath, and S. Manjunath. " Texture features and KNN in classification of flower images", IJCA Special Issue on Recent Trends in Image Processing and Pattern Recognition (PTTIPPR), Vol. 1, pp. 2129, 2010.

[4] Wenjing Qi, Xue Liu, and Jing zhao. Flower classification based on local and spatial visual cues. In CSAE, pages 670-674, 2012.

[5] C. Kanan, G. Cottrell. Robust classification of objects, faces, and flowers using natural image statistics. In Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, pages 2472-2479, 2010.

[6] Angelova A, Zhu S. Efficient object detection and segmentation for finegrained recognition. In Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on, pages 811-818, 2013.

[7] Dayanand G Savakar, Basavaraj S Anami, Recognition and Classification of Food Grains, Fruits and Flowers Using Machine Vision. International Journal of Food Engineering, 01, Jan, 2015.

[8] Fadzilah Siraj, Muhammad Ashraq Salahuddin, Shahrul Azmi Mohd Yusof. Digital Image Classification for Malaysian Blooming Flower. 2010 Second International Conference on Computational Intelligence, Modelling and Simulation, 28-30 Sept. 2010.