Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 51-17 (2018) 280–285

Vegetable Image Retrieval with Fine-tuning VGG Model and Image Hash

Zhaolu. Yang*. Jun. Yue. Zhenbo. Li***. Ling. Zhu******

** Ludong University, Yantai, CO 264025*
China (e-mail: ijoelyang@ 163.com).
*** Ludong University, Yantai, CO 264025*
China (e-mail: yuejuncn@sina.com).
**** China Agricultural University, Beijing, CO 100083 China (e-mail:*
zhenboli@126.com)
***** China Agricultural University, Beijing, CO 100083 China (e-mail:*
1023291660@qq.com)

Abstract: Image descriptors based on activations of Convolutional Neural Networks (CNN) have become dominant in image retrieval due to their discriminative power, compactness of the representation, and the efficiency of search. Fine-tune existing CNN models for image retrieval in specific domain is significant for content-based image retrieval tasks. Inspired by recent successes of CNN with hierarchical features, in this paper, we fine-tuning VGG model to learn features for special vegetable dataset with the classification task. Furthermore, we propose utilizing some PCA Hashing strategies combinate CNN features extracted by the fine-tuned model to improve the performance of special domain CBIR tasks. Our experimental results demonstrate that leveraging the method we proposed can improve the performance of CBIR and the mAP increased by 10 to 20 percent in seam Hash code bits, compared to the model before fine-tuning.

Keywords: Image retrieval, Specific domain, Fine-tune, VGG, CBIR, PCA Hashing

1. INTRODUCTION

Recent years have witnessed great development of content-based image retrieval (CBIR) as shown in Alsmadi et al. (2013). However, many great challenges still exist. One challenging problem in CBIR is the semantic gap between low-level visual features and high level human perceptions as shown in Rafiee et al. (2010). Traditional methods mainly based on low-level features as shown in Zhu et al. (2017), like colour features and texture features, or handcraft features as shown in Yang et al. (2017) and He et al. (2018), like HOG, SIFT, SURF, which usually cannot obtain satisfactory results. Different to traditional methods, convolutional neural network (CNN) can learn hierarchical features as shown in Horn et al. (2017), including high-level features, and has made great contributions in feature extraction for CBIR in recent years. However, in training stage, CNN is always trained by element wise in image space or map the image to one-dimension latent variable. In the retrieval phase, the tasks are almost entirely based on feature extraction and distance metrics. If CNN is trained in this way, it will lead to worse and undesired retrieval performance. Inspired by this, the network is trained by a huge number of data having the same category or different categories and is punished when different categories of samples are identified as more similar to the same category of samples. One of the challenges during the training phase is the lack of large-scale training data sets. Overfitting is a phenomenon where, if there is a problem with the network, the performance on the test set is very poor compared to the perfect performance of the training set.

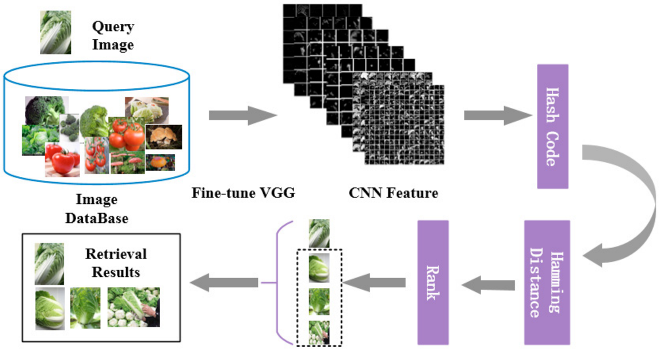

Fig. 1. Retrieval task on Vegetable10 dataset.

Fine-tuning of the network, which mean initialization by a pretrained classification network and re-train for another specific task, is an alternative to a direct application of a pre-trained network. Fine-tuning significantly improves the adaptation ability as shown in Zhang et al. (2014) and Oquab et al. (2014), however, further improve special domain image retrieval task specified training data is required. The first fine-tuning approach for image retrieval is proposed by Babenko et al. (2014), in which a significant amount of manual effort is invested to collect images and label them as specific building classes. They have shown to improve retrieval accuracy, however, their formulation is much closer to classification than to the desired properties of instance

2405-8963 © 2018, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2018.08.175

retrieval. In another approach, Arandjelovic et al. (2016) perform fine-tuning guided by geotagged image databases and, similar to our work, they directly optimize the similarity measure to be used in the final task by selecting matching and non-matching pairs to perform the training.

Our insight is to improve typical feature extraction and achieve the rapid diagnosis for vegetable or special domain decision task. We introduce a batch normalization layer after the convolutional layers and reduce the drop probability in the dropout layer to preserve more convolutional feature. And replace the last two fully connected layers with a convolutional layer and a global pooling layer. Previously, several approaches have been used. These range from fully connected layers to different global pooling layers. We propose a fine-tune model based vgg16 model that has more generalization for most domain image retrieval task. In addition, we propose to employ PCA to reduce dimensions to filter redundant information in VGG convolution features for decreasing algorithm calculating and increasing the speed of retrieval. Furthermore, we conducted an enormous of experiments and compared the search effects of the features of different convolutional layers and fully connected layers and compared them with the original VGG model. The experimental results show that using the features extracted by the fine-tuned VGG model can improve the retrieval performance of CBIR tasks. Our task flow as shown in Fig. 1.

We address unsupervised fine-tuning of CNNs for image retrieval and leverage learning Hash to improve retrieval accuracy and speed and compared using different feature retrieval strategies. We make the following contributions:

(1) We modified and fine-tuned the original VGG model and transfer VGG model to special domain such as Caletch256 datasets, our 10 vegetable datasets and so on by training classification tasks.

(2) We also analysed and compared the relationship between the features extracted from different layers in the model and image retrieval performance.

(3) By combining the CNN features extracted by fine-tune VGG model with different Hash retrieval strategies, the experimental results demonstrate the fine tune feature has better performance in CBIR of special datasets.

2. RELATED WORK

2.1 Search strategy

In recent years, the data on the Internet has rapidly increased, and the data can easily reach hundreds or thousands of dimensions. Therefore, it is infeasible to perform extremely-polarized linear search on such a huge data set. The tree-based and Hash-neighborhood search methods are two popular frameworks, and the Hash method is widely used as a ANN method for large-scale image retrieval due to its advantages in speed and storage by Gong et al. (2016).

Liu et al. (2012) propose KSH, which minimizing the inner product of the Hash code and proves that minimizing the inner product of the Hash code is equivalent to implicit minimizing the Hamming distance makes the problem simpler. In addition, to deal with the nonlinear separability between data, KSH uses a kernel form to construct a Hash function. Xia et al. (2012) designed a two-phase Hash coding learning framework based on deep neural network and proposed a specific Hashing algorithm CNNH. Lai et al. (2015) proposed a new loss function for the problem of learning the Hash function in two stages of the CNNH algorithm and proposed a single-phase learning framework and algorithm based on the deep neural network. Such algorithms based on deep learning improve the effect of supervised Hashing algorithms, but there are also some problems. For example, the design of deep neural networks is complex and the training time is long, which affects the scalability and practicality of the algorithm.

2.2 Feature Extraction

These sorting and searching methods in image retrieval are based on the extraction of artificial visual features from the image, and the artificial features do not need to obtain the similarity of the images, so this may generally affect the effectiveness of these Hashing methods, thereby reducing the image retrieval performance.

With the CNN model of Krizhevsky et al. (2012), training 1.2 million labelled images on the ILSVRC dataset has achieved higher image classification accuracy. In recent years, the deep convolution features have been extensively studied and studied in computer vision and have verified the ability of learning image feature representation successfully. Great breakthroughs have been made in the work. Liu et al. (2014) aim at the inconsistency between low-level visual features and high-level semantics in content-based image retrieval, the traditional distance metrics that are difficult to truly reflect the similarity between images, and so on. This paper proposes a method based on convolutional neural networks and manifold sorting. Image retrieval algorithm. At the same time, Lin et al. (2015) proposed a simple but very effective deep learning framework, adding a hidden layer based on the CNN model to enhance the image feature representation, combined with the Hash function, and achieved better retrieval performance. Although these excellent CNN-based search methods have made great progress today, they do not combine networks with specific fields or data. Therefore, if fine-tuning of models is performed on specific areas of data, they will get better performance.

3. METHODOLOGY

3.1 VGG16 Model

VGG-Net as shown in Simonyan et al. (2014) its outstanding contribution is leveraging a very small convolution in the network. The convolutions of AlexNet and ZFNet by Zeiler et al. (2013) in the first convolutional layer are 11×11 with stride 4 and 7×7 with stride 2, respectively. VGG-Net also prove that two consecutive 3×3 convolutions are equivalent to 5×5 receptive fields and three are equivalent to 7×7. There are two advantages to using three 3×3 convolutions instead of one 7×7 convolution: one, including three ReLU layers instead of one, makes the decision function more discriminative; second, reduces the parameters. The 1×1 convolutional layer is mainly to increase the nonlinearity of the decision function without affecting the receptive field of the convolutional layer. Although the 1×1 convolution operation is linear, ReLU increases the nonlinearity.

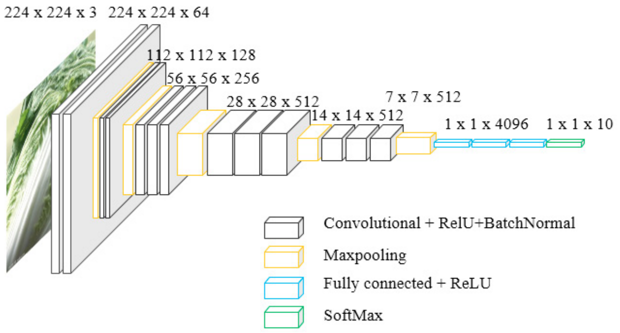

Fig. 2. VGG16 model's structure and parameters.

This paper selects the excellent VGG16 network model to fine-tune our 10 categories vegetable retrieval task. The structure of VGG16 and the parameter settings at each level are shown in Fig. 2. Suppose there is a training dataset with m samples $\{(x(l), y(l)), ..., (x(m), y(m))\}$. For the whole sample, the network overall cost function can be expressed as (1):

$$J(W,b)=\left[\frac{1}{m}\sum_{i=1}^{m}\left(\frac{1}{2}\| K_{w,b}(x^{(i)})-y^{(i)}\|^2\right)\right] + \frac{\lambda}{2}\sum_{l=1}^{n_{l-1}}\sum_{i=1}^{s_l}\sum_{j=1}^{s_{l+1}}(W_{ji}^{(l)})^2 \quad (1)$$

Where $K_{w,b}(x^{(i)})$ is the neural network model, $W_{ji}^{(l)}$ is the connection weight between the *j* th element of layer *l* and the *i* th element of layer *l+1*, and *b* is the bias term of the hidden layer neuron The right side of (1), is a regularization item that reduces the magnitude of the weight, which can prevent overfitting, and λ adjusts the relative importance of the two terms before and after the cost function. Solving the minimum value of (1) adopts the well-known batch gradient descent optimization algorithm, and when calculating the partial derivative of $J(W,b)$ for *W* and *b*, the reverse conduction algorithm is used.

3.2 Fine-tuning VGG

Fine-tuning means to modify the network to adapt to our own specific new tasks. In general, we need to do our own direction. For example, in the identification and classification of certain specific fields, it is difficult for us to get a lot of data like ImageNet. After all, it is tens of millions of image databases, usually we may only get a few thousand or tens of thousands of images in a specific field, such as the vegetables in this article, or other clothes, biological species, and so on. In this case, retraining a new network is more complicated, and the parameters are not good enough to adjust, and the amount of data is not enough, so fine-tuning is an ideal choice. There is a great difference between the target task's image set and pre-trained image set, regardless of the number of categories or image styles. In the retrieval task of the target image set, the visual features of the image are directly extracted by the pre-trained CNN model. It is difficult to achieve optimal performance. Therefore, to make the pre-trained CNN model parameters more suitable for the feature extraction of the target image set, the image of the target image set is used to fine-tune the parameters of the pre-trained CNN model. The fine-tuning process is as follows:

1) As in the pre-training process, each image from the target image library is first adjusted to 256 × 256, and then the sub-blocks of the image 224 or their mirror images are randomly extracted as CNN input.

2) Assume that the number of categories of the target image set is C, change the number of outputs of SoftMax layer at the last layer of the CNN model from 1000 to C, and randomly initialize the last layer of network parameters. For other layer network parameters, initialized by the weights obtained from the pre-trained VGG model.

3) The stochastic gradient descent (SGD) was used to adjust the parameters of the entire network. During the training, the momentum was set to *0.9*, and the weight attenuation coefficient was set to *0. 0005*. For all layers, ReLU as a nonlinear activation function was used, and add the Batch Normalization layer, which different from the original network.

3.3 Feature Selection

Feature selection is still a crucial part of image retrieval. Through the introduction of Section 3.2, we generalize the general features of VGG learning through the finetune way, so that the expressed features can better adapt to our search tasks. In this section, we will complete the feature extraction before and after the VGG model Fine-tuning, and further explore whether fine-tuning is helpful to the retrieval effect, and which layer of feature extraction best performance.

Table 1. mAP for different layer features of VGG and Fine-Tune VGG on Vegetable10

Features & Hash		Hash			
		CBE-opt	ITQ	PCA-RR	SKLSH
VGG	Conv5	0.398	0.419	0.390	0.276
	FC6	0.746	0.742	0.728	0.607
	FC7	0.726	0.747	0.719	0.720
	FC8	0.503	0.544	0.591	0.718
FTVGG	Conv5	0.250	0.266	0.265	0.234
	FC6	0.579	0.582	0.592	0.375
	FC7	**0.787**	**0.865**	**0.904**	**0.886**
	FC8	0.827	0.820	0.821	0.784

For retrieval tasks, or other learning tasks, when we use the pre-trained model to extract features, Is the final layer necessarily the final one? VGG16 contains three full-connection layers, FC6, FC7 and FC8. To this end, we extracted the features of the last layer of convolutional layers and the features of the three fully connected layers of VGG model and analysed and compared the effects of utilizing different layer of features on the Mean Average Precision (mAP) of image retrieval. Table 1 shows that the FC7 layer performs best among the compared network layers. The data in the table also shows that the performance of FC7 and FC8 after the fine-tuning model has improved, as shown in Table 2 below, but the performance of CONV5 and FC6 layer was different from expectations.

3.4 Combination of Fine-tune VGG and Hash code

In the image retrieval, if the features of the image extraction are not outstanding at the beginning, then a high level of Hash coding strategy to perform the coding search will not necessarily achieve good results. Therefore, this paper proposes a combination of fine-tune CNN feature and the best level of Hashing strategy. We use the Fine-tune VGG16 model network model to perform feature extraction on the target image set, obtain the deep convolutional representation of the image, and then represent these separately. The Hash code is encoded using CBE by Yu et al. (2014) SH by Weiss et al. (2008), SpH by Heo et al. (2015), ITQ and PCA- RR by Gong et al. (2011), and SKLSH by Raginsky et al. (2009), and the Hash code is obtained. Finally, fast retrieval is performed.

4. EXPERIMENTAL ANALYSIS

4.1 Image sets and evaluation indicators

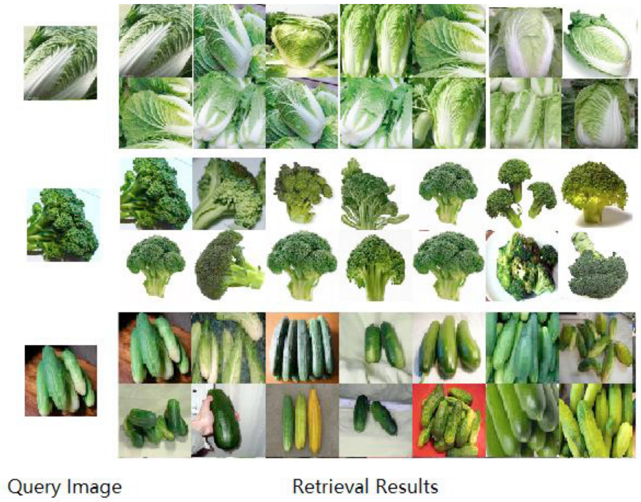

Fig. 3. Query images and retrieval results for different types of vegetables.

The performance of the CNN visual features of different layers in the image retrieval task is compared and analysed in the image libraries that they have collected and photographed. Our vegetable10 image library consists of a total of 10 classes and 10,000 images. Each class contains 1000 image samples. A random selection of 200 pictures from each class was used as a query image. Image retrieval generally uses precision, recall and Mean average precision (mAP) as evaluation indicators.

$$percision = (a/b) \times 100\% \quad (2)$$

$$recall = (a/c) \times 100\% \quad (3)$$

$$mAP(Q) = \frac{1}{|Q|}\sum_{j=1}^{|Q|}\frac{1}{m_j}\sum_{k=1}^{m_j}\Pr ecision(R_{jk}) \quad (4)$$

Where a is the number of correct results in the returned result, b is the number of results returned, and c represents the total number of related results in the system, $Q = \{e_1, e_2, ...e_m\}$, R_{jk} indicating the sorting result when the d_k element is retrieved.

Table 2. CNN feature from FC7&FTFC7 for mAP between different bits on Vegetable 10 dataset

Hash bits		Number of bits				
		16	32	64	128	256
VGG FCLayer7	CBE-rand	0.133	0.280	0.443	0.587	0.701
	CBE-opt	0.178	0.360	0.508	0.685	0.726
	ITQ	**0.357**	**0.537**	**0.639**	**0.705**	**0.747**
	PCAH	0.249	0.237	0.214	0.198	0.229
	SH	0.280	0.277	0.350	0.438	0.545
	SKLSH	0.061	0.121	0.287	0.465	0.720
	PCA-RR	0.257	0.403	0.557	0.640	0.719
	SpH	0.189	0.247	0.347	0.435	0.477
Fine-Tune FCLayer7	CBE-rand	0.389	0.503	0.654	0.761	0.877
	CBE-opt	**0.473**	0.588	0.726	0.789	0.787
	ITQ	0.397	0.576	**0.745**	0.823	0.865
	PCAH	0.451	0.430	0.392	0.347	0.320
	SH	0.461	0.585	0.703	0.788	0.838
	SKLSH	0.221	0.482	0.725	0.813	0.886
	PCA-RR	0.455	**0.641**	0.711	**0.829**	**0.901**
	SpH	0.411	0.432	0.471	0.541	0.557

Table 2 demonstrates the average accuracy of the fully connected layer features before and after tuning with the vgg model under different bit Hash codes on our 10 vegetable datasets. Bold fonts are the best method under different encodings. From the table we can conclude that under the characteristics extracted by the original vgg model, ITQ shows excellent performance in various bits encoding. After fine-tuning, PCA-RR achieved very good performance in 32bit, 128bit and 256bit, and ITQ can still stand out in 64-bit encoding. However, mAP increases with the increase in the number of coded bits regardless of whether it is on the original VGG and on the fine-tuned VGG.

4.2 Experimental results

In this section, several Hash strategies for ITQ, PCA-H, CBE, SPH and SH are described in our vegetable image sets and Caltech256 dataset. We use a HABIR Hash Image Retrieval Toolbox of MATLAB, created by Yuan et al. (2014), which integrates the classic Hash method and the benchmark framework for unsupervised Hash methods in recent years. We use different Hash and these previous best-level image Hash retrieval strategies to compare and employ the commonly used five Hash code bits 16, 32, 64, 128 and 256 to perform experiments on the Precision - Recall standard.

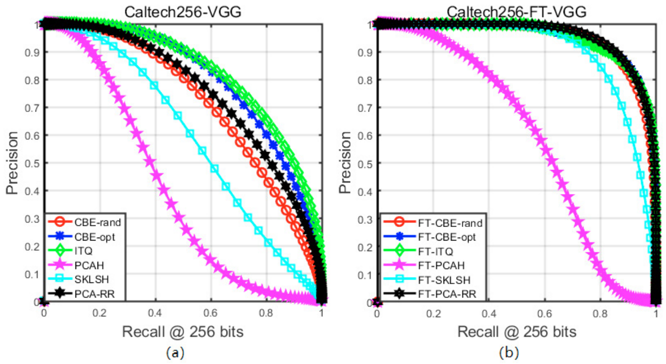

Fig. 4. (a) In the 256-bit encoding, the Precision-Recall curve using original VGG model (b) Under 256-bit coding, the Precision-Recall curve using Fine-Tune VGG model.

The Precision-Recall curves of (c) in Fig. 4, 5 and Fig. 6 illustrate that most of the strategies using our proposed method have significantly improved the retrieval performance compared to the VGG model before fine-tuning, although the PCAH improvement is relatively small. The mAP is a single-valued indicator that reflects the performance of all relevant data methods. The higher the level of relevant data retrieved using this method, the higher the mAP may be. The experimental results of the method in mAP are shown in Fig. 5(d) and Fig. 6(d), and the mAP is continuously improved as the encoding continues to be lengthy.

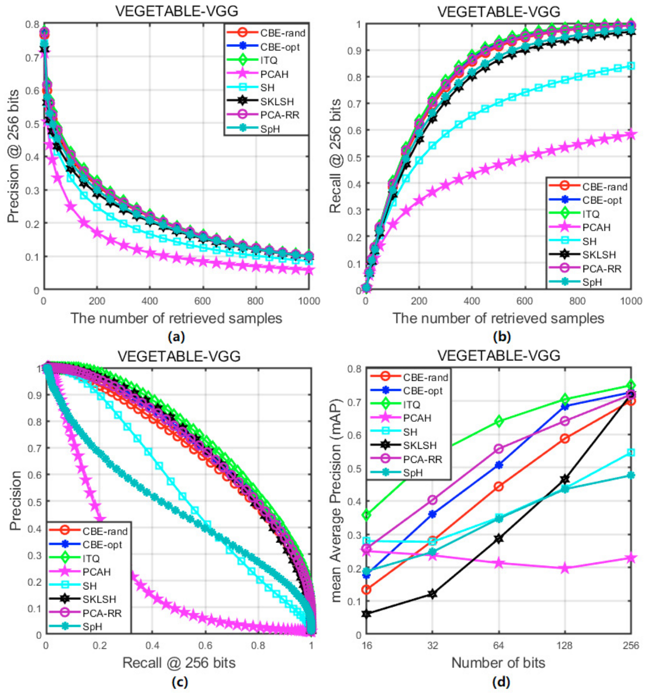

Fig. 5. VGG model (a) In 256-bit encoding, Precision rate of the curve with the number of returned samples (b) In 256-bit coding, Recall rate of the curve with the number of returned samples (c) Precision-Recall curve (d) The mAP varies with the length of the code.

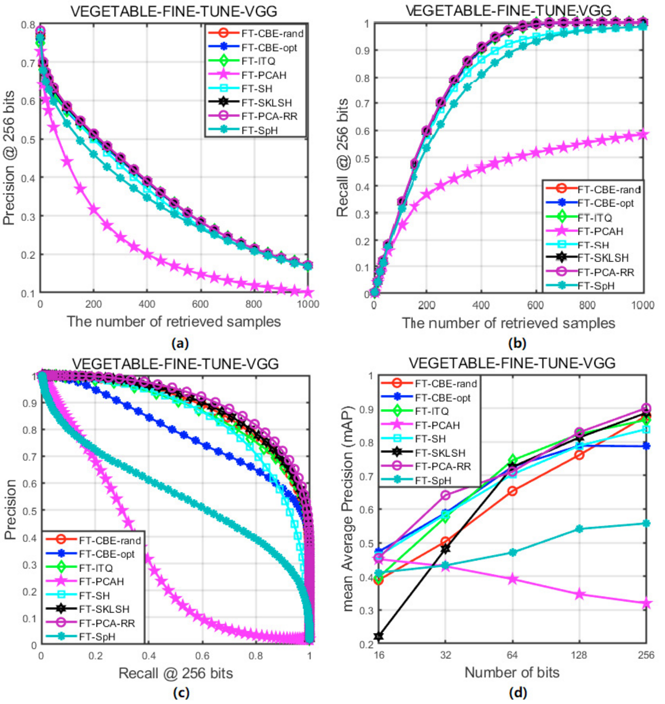

Fig. 6. FT-VGG model (a) In 256-bit encoding, Precision rate of the curve with the number of returned samples (b) In 256-bit coding, Recall rate of the curve with the number of returned samples (c) Precision-Recall curve (d) The mAP varies with the length of the code.

Fig. 5 and 6 show the search results by combining CNN features and different Hashing strategies before and after fine-tuning the VGG16 model. It can be clearly seen from the comparison of the pictures that after using the fine-tuning feature, the curve of most methods still has obvious improvement, especially the precision of the precision Recall and the average precision of different code lengths. Although the relationship between accuracy rate, recall rate, and return sample size does not change significantly, we can still determine the curve change trend by observing the intersection of the curve and the grid. In addition, from the comparison of these methods, PCA-ITQ has been performing well, and the improvement of SKLSH in the characteristics of fine-tuning model is more obvious, but PCAH and SpH are relatively poor whether before or after fine-tuning.

5. CONCLUSIONS

In the task of image retrieval for a specific area, this paper proposes an effective method that combines the dimensionality reduction of CNN features extracted by fine-tune model and PCA for Hashing strategy. First, the VGG16 network model was modified and fine-tuned to fit special domain images by classification task, and then extract the deep feature from the image set. Then we experimentally analysed and compared the effects of image features of different network layers on the performance of retrieval tasks and concluded that the FC7 layer has better performance.

Finally, we compared the mAP of the image retrieval of the fine-tuning features on different data sets and concluded that the fine-tuned CNN visual features performed better in the CBIR task.

Although this article has drawn some conclusions through experimental comparison and analysis, we still have some limitations in our work. There are still many tasks that need to be done in the future. For example, we can conduct experiments on more uncommon data sets. In addition, we only compared the retrieval performance between the last layer of convolutional layers and the fully connected layer. Whether other convolutional layers or pooling and activation layers will perform better, we will continue to explore in future work.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the contribution to this research provided by the sponsors of National Natural Science Foundation of China (61472172), Key Industry and Key Technology Project of province (2016CYJS03A02-1) and City's key R&D projects (2017ZH057).

REFERENCES

Alsmadi, M. K. (2017). An efficient similarity measure for content based image retrieval using memetic algorithm. Egyptian Journal of Basic & Applied Sciences, 4(2).

Babenko, A., Slesarev, A., Chigorin, A., & Lempitsky, V. (2014). Neural codes for image retrieval. 8689, 584-599.

Gong, Y., & Lazebnik, S. (2011). Iterative quantization: A procrustean approach to learning binary codes. IEEE Conference on Computer Vision and Pattern Recognition (Vol.42, pp.817-824). IEEE Computer Society.

Gong, ZT. REN, GX. et al. (2016). An image retrieval method based on a convolutional neural network and Hash coding. CAAI Transactions on Intelligent Systems,11(3):391-400.

He, T., Wei, Y., el al. (2018). Content Based Image Retrieval Method Based on SIFT Feature. 2018 International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS), pp. 649-652.

Heo, J. P., He, J., He, J., Chang, S. F., & Yoon, S. E. (2015). Spherical Hashing: binary code embedding with hyperspheres. IEEE Transactions on Pattern Analysis & Machine Intelligence, 37(11), 2304-2316.

Horn, Z. C., Auret, L., Mccoy, J. T., Aldrich, C., & Herbst, B. M. (2017). Performance of convolutional neural networks for feature extraction in froth flotation sensing. IFAC-PapersOnLine, 50(2), 13-18.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. International Conference on Neural Information Processing Systems (Vol.60, pp.1097-1105). Curran Associates Inc.

Lai, H., Pan, Y., Liu, Y., & Yan, S. (2015). Simultaneous feature learning and Hash coding with deep neural networks. 3270-3278.

Lin, K., Yang, H. F., Hsiao, J. H., & Chen, C. S. (2015). Deep learning of binary Hash codes for fast image retrieval. Computer Vision and Pattern Recognition Workshops (pp.27-35). IEEE.

Liu, S., Cui, P., Zhu, W., Yang, S., & Tian, Q. (2014). Social Embedding Image Distance Learning. ACM International Conference on Multimedia (pp.617-626). ACM.

Liu, W., Wang, J., Ji, R., & Jiang, Y. G. (2012). Supervised Hashing with kernels. Computer Vision and Pattern Recognition (Vol.157, pp.2074-2081). IEEE.

Oquab, M., Bottou, L., Laptev, I., & Sivic, J. (2014). Learning and Transferring Mid-level Image Representations Using Convolutional Neural Networks. IEEE Conference on Computer Vision and Pattern Recognition (pp.1717-1724). IEEE Computer Society.

Rafiee, G., Dlay, S. S., & Woo, W. L. (2010). A review of content-based image retrieval. International Symposium on Communication Systems Networks and Digital Signal Processing (Vol.1, pp.775-779). IEEE.

Raginsky, M. (2009). Locality-sensitive binary codes from shift-invariant kernels. Advances in Neural Information Processing Systems, 1509-1517.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. Computer Science.

Weiss, Y., Torralba, A., & Fergus, R. (2008). Spectral Hashing. International Conference on Neural Information Processing Systems (Vol.282, pp.1753-1760). Curran Associates Inc.

Xia, R., Pan, Y., Lai, H., Liu, C., & Yan, S. (2012). Supervised Hashing for image retrieval via image representation learning. AAAI Conference on Artificial Intelligence.

Yang, Z., Yue, J. et al. (2017). Automated and low-cost reconstruction method for cactus 3D phenotyping. IAEJ, 26(4): 370-379.

Yu, F. X., Kumar, S., Gong, Y., & Chang, S. F. (2014). Circulant binary embedding. Computer Science, 946-954.

Yuan, Y., Lu, X., & Li, X. (2014). Learning Hash functions using sparse reconstruction.

Zhang, N., Donahue, J., Girshick, R., & Darrell, T. (2014). Part-based r-cnns for fine-grained category detection, 8689, 834-849.

Zeiler, M. D., & Fergus, R. (2014). Visualizing and Understanding Convolutional Networks. European Conference on Computer Vision (Vol.8689, pp.818-833). Springer, Cham.

Zhu, L., Li, Z., Yang, Z., Li, C., Wu, J., & Yue, J. (2017). Internet eggplant image retrieval method and system based on mixed features. Transactions of the Chinese Society of Agricultural Engineeri.