Flower Classification with Few Training Examples via Recalling Visual Patterns from Deep CNN

Kevin Lin (林可昀) Huei-Fang Yang (楊惠芳) Chu-Song Chen (陳祝嵩)
Academia Sinica, Taipei, Taiwan

Abstract

*A recent trend of research has been focused on generalizing/transferring the CNN features pre-trained on a large-scale dataset (*e.g.*, the ImageNet) to perform a new task in another domain. Though superior results have been obtained by adapting the CNN features to another domains, how to fine-tune a deep CNN with very few training samples remains a problem. In this paper, we propose a framework that can enrich the training examples for fine-tuning a CNN. The central idea is to recall similar patterns from the pre-trained model and include these recalled images in re-training the network. We conduct experiments on the Oxford 17 category and 102 category flower datasets. Experimental results show that enriching the training data improves the performance of the fine-tuned network. Our method also demonstrates superior performance over other state-of-the-art approaches.*

Flower Classification with Few Training Examples via Recalling Visual Patterns from Deep CNN

Kevin Lin (林可昀) Huei-Fang Yang (楊惠芳) Chu-Song Chen (陳祝嵩)
Academia Sinica, Taipei, Taiwan

Abstract

*A recent trend of research has been focused on generalizing/transferring the CNN features pre-trained on a large-scale dataset (*e.g.*, the ImageNet) to perform a new task in another domain. Though superior results have been obtained by adapting the CNN features to another domains, how to fine-tune a deep CNN with very few training samples remains a problem. In this paper, we propose a framework that can enrich the training examples for fine-tuning a CNN. The central idea is to recall similar patterns from the pre-trained model and include these recalled images in retraining the network. We conduct experiments on the Oxford 17 category and 102 category flower datasets. Experimental results show that enriching the training data improves the performance of the fine-tuned network. Our method also demonstrates superior performance over other state-of-the-art approaches.*

1. Introduction

Flower classification in the wild is a challenging task due to the high variation of shapes, color distributions, lighting condition, and pose deformation. What makes it more difficult in real application scenarios is that only few training data can be collected as usual for a flower class. For example, when one picture of flowers or two are taken by a mobile phone, a user would like to have a flower recognizer determining the flowers of the same category in the future. Only few training examples are available, and the foreground region of the flower is generally unknown. As such, a powerful classifier that can be learned from domain-specific image representations with limited training exemplars of the flowers becomes necessary.

Convolutional neural networks (CNNs) [23] are capable of learning rich mid-level image representations that have been proven to be effective for many vision recognition tasks, such as image classification [20, 40], digit recognition [8], and pedestrian detection [34]. A recent trend of research has centered around generalizing/transferring the CNN features pre-trained on a large-scale dataset (*e.g.*, the ImageNet [9]) to perform a new task in another domain (*e.g.*, the PASCAL VOC [11]) [7, 16, 30].

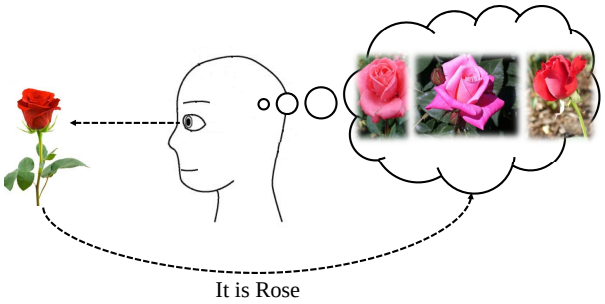


Figure 1: Our main idea to enrich the training data is through associating the new stimulus to the learned knowledge. During learning, the new information is integrated with the previous experiences.

Though superior results have been obtained by adapting the CNN features to another domains, a sufficiently large amount of training data is still a necessity. Oquab *et al*. [30] and Girshick *et al*. [16] used all the available training examples from the PASCAL VOC, together with multiple hypotheses generated for each training sample, to fine-tune a deep CNN pre-trained on ImageNet. However, in many visual recognition domains, one may hope to train a classifier with only a few samples; this is especially true in the flower recognition task. How to fine-tune a deep CNN with very few training examples thus becomes an issue for such applications.

To address this problem, we propose to enrich the training examples through recalling similar patterns from the network and use the enriched training data to fine-tune the CNN. This is inspired by the theory that during learning, the new information can be assimilated or linked to the previous experiences [2] (see Figure 1). Therefore, the previously learned CNN features can be considered as knowledge coming from previous experiences. When new information comes, it is integrated with the relevant piece of the stored information. In other words, our approach is to reuse the patterns recalled from the network to refine the connection strength between neurons in the training. More precisely, we treat those annotated samples from the target do-

main as queries to the trained CNN to retrieve relevant exemplars from the large-scale datasets. To this end, we fine-tune a deep CNN on the enriched training data, including the annotated samples and relevant unlabeled exemplars, in a semi-supervised manner. In sum, our method is with the following characteristics: (1) The deep CNN can learn image representations from limited training samples; and (2) recalling similar patterns from a pre-trained CNN is a simple, yet effective, way to enrich the training data.

This paper is organized as follows: We briefly review the related work of flower classification and deep CNN in Section 2. We elaborate on the details of our method in Section 3. Finally, experimental results are provided in Section 4, followed by conclusions in Section 5.

2. Related Work

2.1. Flower Classification

Flower classification poses a unique challenge task because most flower categories have a significant visual similarity, indistinguishable on color (or shape) alone. Discriminating one flower from another mainly rely on a combination of different cues, such as shape, color, and texture patterns. One simple way to combine features is to learn the weights for individual cues [28]. Other methods for feature combination include the boosting approach, the Frequent Local Histograms (FLHs) [14], and Multiple Kernel Learning (MKL) [15]. Recently, due to the emergence of sparse representation, the Sparse Representation-based Classification (SRC) [36] has also been applied to flower categorization. Yang *et al*. [38] proposed to use the Fisher discrimination dictionary learning (FDDL) model in learning structured dictionaries, which yields impressive performance on flower classification.

2.2. Deep CNN

Krizhevsky *et al*. [20] demonstrated the outstanding performance a deep CNN can achieve on performing the 1000-class classification task in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC). This performance leap is attributed to training a CNN to learn image representations on more than one million images. The CNN-based image representations have also demonstrated superior performance over handcrafted image descriptors including the SIFT [25] and Fisher Vector (FV) [31].

Given that the deep CNN architectures learn powerful rich image descriptors, transferring these CNN-based representations to other smaller datasets has been a focus of recent research. The strategy is to have the CNNs to learn good representations on large scale data in either an unsupervised [22, 26] or a supervised [7, 10, 16, 30] manner. Once the CNNs have learned representations of visual features, these representations can be transferred to another domains, with or without adapting them to represent domain-specific features. Chatfield *et al*. [7] empirically evaluated the performance of transferred deep representations and showed that the fine-turned representations yielded better performance than the nonadapted ones.

Most of the existing work on transferring the CNN representations to other domains has focused on the visual recognition tasks using PASCAL VOC, Caltech-101, and Caltech-256 datasets. In the domain of flower categorization, it still relies on the handcrafted features [14, 37, 38, 39] till a recent work proposed by Razavian *et al*. [32] that used the CNN features for the flower recognition. However, they regarded the pre-trained CNN a feature extractor and did not fine-tune the network for the new task. This is a main difference between their work and ours.

2.3. Learning with Few Training Examples

The ability of learning with few examples has been demonstrated in humans [4]. This is attributed to the ability of synthesizing and learning new object from prior knowledge. Several related works [19, 12, 27, 3] mimic this learning process, taking the advantage of knowledge from previous learned categories to learn new classification task. Li *et al*. [12, 13] proposed the one-shot learning via Bayesian approach. The proposed posterior model is adapted by updating the prior of one or few observations. Thorsten [19] and Miller *et al*. [27] transferred the parameter from previous learned model to the new one for learning new objects. Bart and Ullman [3] replaced the features from the learned categories with similar features taken from the new objects. Other works [17, 35, 41] tackle this problem using semi-supervised learning, where the training data is expanded by searching supplemental images from other sources (*e.g*., Internet). The classifier is obtained by training with the augmented dataset.

While a pre-trained deep CNN is approximated to a storage of learned knowledge (or human brain), we can take the advantage of deep CNN to learn new objects from few training samples. Instead of previous approaches [19, 12, 27, 3], we fine-tune deep CNN to associate new information with the previous learned knowledge. Unlike previous semi-supervised methods [17, 35, 41] that require external data sources, our method follows the self-training principle that enriches the training samples by enhancing the relevant visual patterns recalled from the deep CNN storage.

3. Method

Figure 2 shows the workflow of our proposed method. Our method includes three main components. The first component is the supervised pre-training of a CNN on a large-scale dataset. The second is enriching the training data for the new task by recalling similar patterns from the

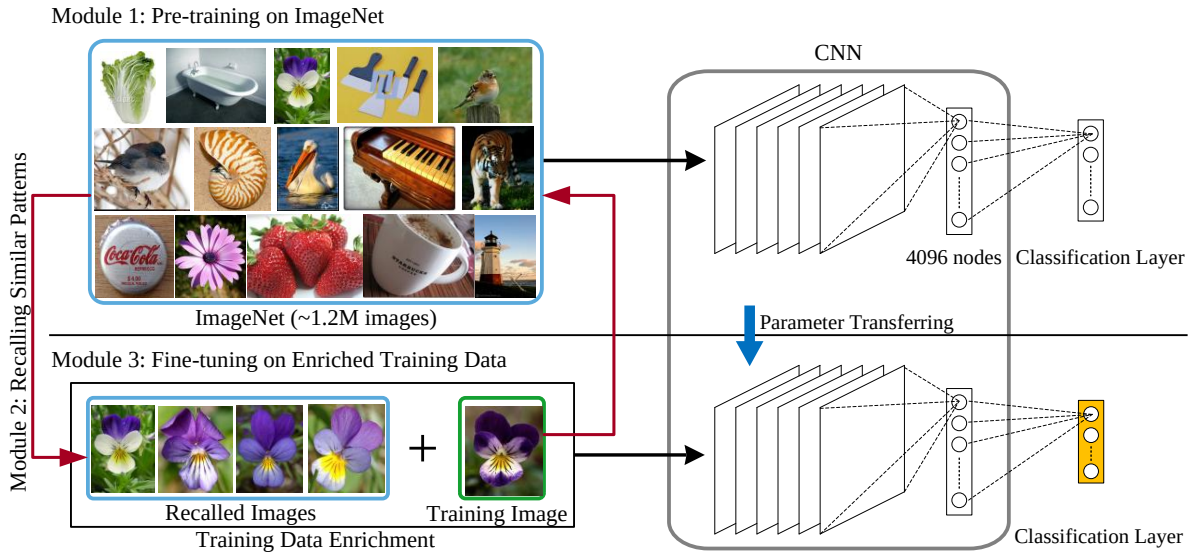


Figure 2: Overview of our proposed framework for fine-tuning a deep CNN with few training samples. Our method consists of three main components. The first component is the supervised pre-training of a CNN on the ImageNet dataset. The second is enriching the training data for the new task by recalling similar patterns from the network. The third fine-tunes the CNN to obtain new image representations for the new task. The final layer of the pre-trained is replaced with a new layer that represents the labels in the new domain.

network. The third fine-tunes the CNN to obtain new image representations for the new task.

In the first component, we use the pre-trained deep CNN model proposed by Krizhevsky *et al*. [20] from Caffe CNN library [18]. The subset of ImageNet used for training the deep CNN contains more than 1.2 million images. In parallel to the CNN training purpose, we also use it as a large database or corpus for image retrieval. Hereafter, we use $\mathbf{\Gamma}$ to denote this database.

3.1. Training Data Enrichment

Pseudo relevance feedback (PRF) has been shown to be effective for information retrieval [5, 21, 24]. The rationale behind PRF is that when a search for a given query is performed, the top k ranked retrieved documents are relevant. These most relevant documents can then be used to expand the query terms in order to retrieve more relevant or similar documents.

Our method of associating the new information from the target domain to the knowledge learned from previous experiences share similar spirits to the PRF: we "expand" the size of training examples by including the top ranked retrieved images. While similar to PRF in the concept of data expansion, our method is to integrate the new information from the target domain with the previous experiences. It aims at making deep CNN learn new image representations from the expanded samples for a new task. This is different from the PRF in information retrieval that the retrieved documents are mainly for reformulating the original query to improve retrieval accuracy.

Our idea to enrich the training samples for the new task is to recall similar patterns from the pre-trained network. For this, we consider measuring the similarity between two images based on their signatures of layer FC_7 of CNN (referred to as the CNNFC_7 feature). To extract the features, we first resize an RGB image to 256×256 pixels. Then, we use its mean-subtracted 227×227 center patch as input to the pre-trained model. Through a forward propagation, we obtain a $d = 4,096$-dimensional feature for an image.

Given a few training images $\mathbf{\Pi} = \{I_1, I_2, \cdots, I_n\}$ from the new domain. Let q_i $(i = 1 \cdots n)$ denote the d-dimensional CNNFC_7 feature vectors extracted for the i-th training images. Let p_j denote the CNNFC_7 feature vectors of the j-th image in the database $\mathbf{\Gamma}$. We define the level of similarity between images I_i and the j-th image of $\mathbf{\Gamma}$ as the Euclidean distance between their corresponding features,

$$s_j^i = \|q_i - p_j\|. \quad (1)$$

The smaller is the Euclidean distance, the higher level is the similarity of the two images.

For each training image I_i, we select its top k ranked images from the database $\mathbf{\Gamma}$ based on the similarity levels. These are called the recalled images from the corpus $\mathbf{\Gamma}$. Both the training images in $\mathbf{\Pi}$ and the recalled images in $\mathbf{\Gamma}$ are then included in the training samples during the fine-tuning process that will be introduced in the following.

3.2. Fine-tuning

To adapt the pre-trained model to the new task, we replace the final layer of the network with a new classifica-

tion layer that represents the labels in the target task. Other layers of the CNN architecture remain unchanged. We use the stochastic gradient descent (SGD) to train the network, maximizing the multinomial logistic regression objective. The learning rate is set to 0.0001 and remains unchanged during the whole fine-tuning process.

We use the enriched training data depicted above as the positive examples, and randomly select images, which do not belong to any flower classes, from ImageNet as the negative samples. A mini-batch of size 128, including 32 positive and 96 negative samples, is used in each SGD iteration.

4. Experimental Results

We conduct experiments on two challenging datasets, the Oxford 17 category and 102 category flowers data. We focus on these two datasets not only because the amount of training data is scarce, but also because the flower recognition poses a challenging task in the real world applications.

We start with introducing the datasets and then present our experimental results as well as comparison to other methods for each dataset. Next, we analyze how the fine-tuning process affects the network parameters. Finally, we report the computational complexity of our method.

4.1. Datasets

The Oxford 17 Category Flower Dataset [28] contains 17 categories and each class consists of 80 images, resulting in a total of 1,360 images. The dataset is split into the training (40 images per class), validation (20 images per class), and test (20 images per class) sets.

The Oxford 102 Category Flower Dataset [29] contains 102 categories and each class consists of 40 $\sim$ 258 images. This dataset contains a total of 8,189 images. Similar to the 17 category dataset, this larger one is also split into the training (10 images per class), validation (10 images per class), and test (a total of 6,149 images) sets.

Although both datasets is predefined into different sets, we use only the training and test sets; the validation one is not included in our experiments. During the recall process to retrieve images from ImageNet, we randomly pick n images per class from the training set. The test set is used for performance evaluation. In both datasets, the images in each category exhibited high variations in scale, pose, and light. Besides, some categories are of the images with large variations within the class. Figure 3 shows some sample images from both datasets.

Note that while the segmentations for both datasets are available, we use only the raw images in our experiments. Therefore, the results obtained by our method do not rely on precise pre-segmentations, making our approach more suitable for practical applications.

Figure 3: Sample images from the flower datasets. The top row are the daisy images from the 17 category flower dataset, and the bottom the rose images from the 102 category flower dataset.

4.2. Results on 17-Category Dataset

Recalled Images from ImageNet One of our main features is the association of the new stimuli from the target domain (*e.g.*, flower datasets) to the knowledge learned from the previous experiences (*e.g.*, ImageNet). This association process aims at forming a new set of training samples in order for the machine to learn to perform the new task. The new stimulus is regarded as the query image that is used to retrieve highly relevant images from the learned knowledge base. Figure 4 shows the query images (the daisy and sunflower) from the Oxford 17 category flower dataset and their corresponding top 10 ranked recalled images from ImageNet. As can been seen, though the query and retrieved images are from two different domains, images from the target domain can successfully be associated to the learned images. Compared to using only images from the target domain, our association of new images to the previous learned images can increase the size of training samples that will help the pre-trained CNN learn new image representations during the fine-tuning process.

Classification Performance v.s. Different Sample Sizes Used in Fine-tuning The CNN's ability to learn to perform the new task may rely on the size of training examples from the target domain. We conduct experiments by varying the number of samples from the target domain as well as the number of recalled images to investigate how the CNN performance changes. More specifically, we aim at answering the following questions:

1. Can only a few samples from the target domain improve the CNN performance?

2. Can the learned knowledge from previous experiences help the CNN learn to perform the new task?

To answer the former question, we first use only 1 or 5 images per class from the target domain to fine-tune the CNN. As shown in Figure 5 and Table 1, the classification accuracy obtained by using 5 images per class is increased to 92.10%, compared to a 87.25% obtained by using 1 image per class, leading to a 4.85% gain on the classification

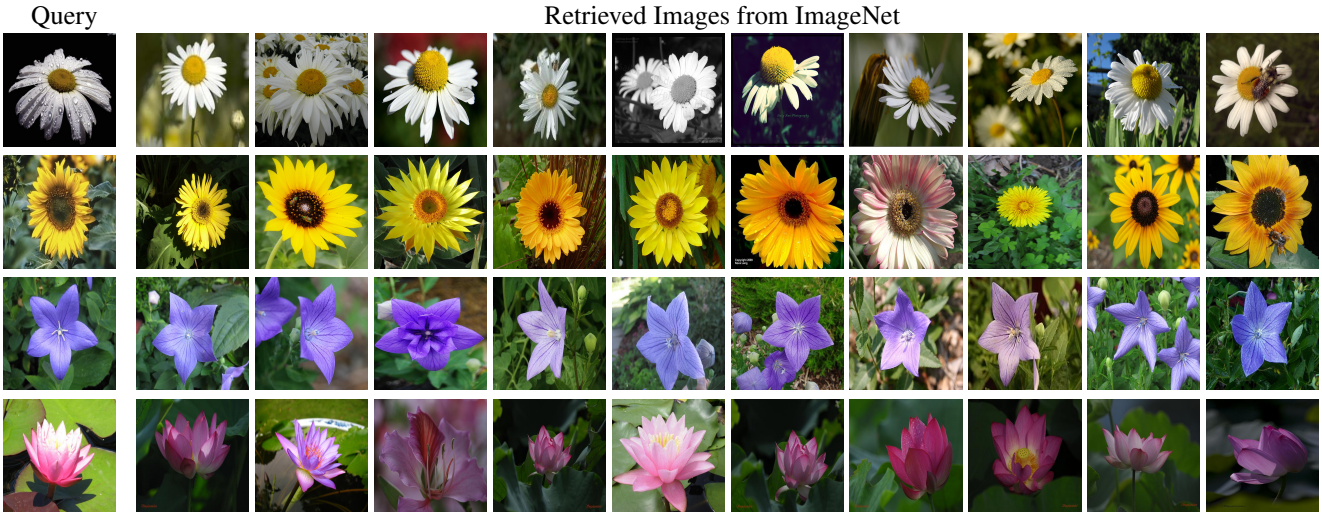


Figure 4: Query images from the Flowers dataset and their corresponding top 10 ranked images retrieved from ImageNet. The first two query images are taken from the 17 category dataset; the bottom two from the 102 category one. Visually, the query and retrieved images are similar in appearances, shapes, or both although they are from two very different domains.

accuracy. Further increasing the number of images per class to 40 from 5, we obtain a 3.13% improvement on classification accuracy. As expected, the CNN performance is increasingly improved as the number of images per class from the target domain increases. Note that even with only a few training samples from the target domain used in fine-tuning, the network can still gain a considerable amount of performance improvement.

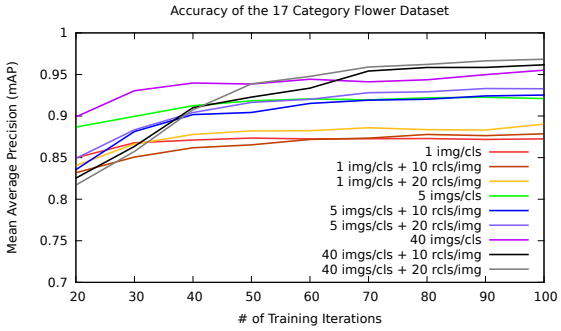
We further include the recalled images from ImageNet, together with the images from the target domain, in the fine-tuning process. In this experiment, we first use 1 or 5 images per class from the target domain as the query samples to retrieve top k ranked images from ImageNet for each query. The performance of CNN is shown in Figure 5. Comparing the gray curve against the black and purple ones, we can see that when the same number of images from the target domain is used, performance of the CNN is increased as more relevant images from ImageNet are included in the fine-tuning. More precisely, adding 20 recalled images per query sample in the fine-tuning can improve the performance by a margin of 1.77%, 1.18%, and 1.81% compared to their counterparts (with 1, 5, 40 images per class) where no previous experiences are involved (see Table 1). This suggests that the learned knowledge from previous experiences can help the CNN adapt its image representations to the new task.

Performance Comparison to Other Methods We compare our method to the state-of-the-art methods, as shown in Table 1. The existing methods aim at learning discriminant models from several cues (*e.g.*, HSV, HOG, color, shape, and texture) for the classification task, based on the Multiple Kernel Learning (MKL) [15], the Frequent Local Histograms (FLHs) [14], or the Fisher Discrimination Dictionary Learning (FDDL) [38] approach. As these methods used handcrafted or dimension-reduction features for image classification, to investigate to what extent the pre-trained CNN features would achieve, we use the FC_7 of the pre-trained network as visual features and train a one-vs-rest SVM classifier for each class. This experiment involves no fine-tuning process and is denoted by CNN-SVM in Table 1.

The CNN-SVM achieves a classification accuracy of 91.52% using 40 images per class in training. Our proposed method outperforms the CNN-SVM baseline when 5 images per class are used in fine-tuning. The best accuracy achieved by our method is 96.84% (with 40 training images), which is favorable against most approaches though slightly worse than the Yang *et al*. [38] method. However, note that our method attains the accuracy based on un-segmented images, while the method in [38] assumes that the images are well segmented. Besides, we show that by using only few training images (*e.g.*, one or five), the accuracies of 89.02% and 93.28% can be achieved.

4.3. Results on 102-Category Dataset

Recalled Image from ImageNet The bottom two rows in Figure 4 show the query images (*i.e.*, the Balloon flower and lotus) from the 102 category dataset and the top ranked recalled images. Though the 102 Category dataset contains more image classes and poses a more challenge task than the 17 category one, the recall procedure is capable of retrieving the highly relevant images. However, due to close similarity between flower classes, some images may be wrongly recalled, for example, the retrieved image at rank 3 for the query image of lotus.



∗ cls, img, and rcl denote class, image, and recall, respectively.

Figure 5: Comparison of the classification accuracy on the Oxford 17 category Flower dataset with respect to different numbers of images used in the fine-tuning process. As expected, the classification accuracy increases as we increase the numbers of images per class from the flowers dataset used in the fine-tuning process (comparing the purple curve to the red and green). The classification performance is further improved when more relevant images from ImageNet are included in fine-tuning (the gray curve).

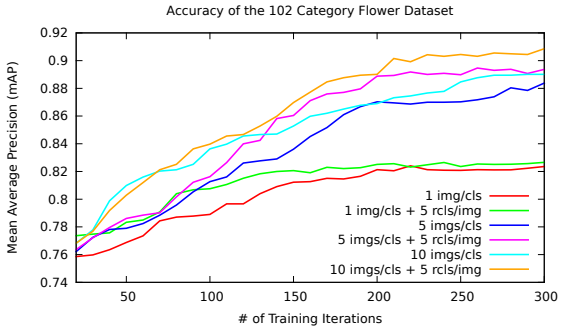
Table 1: Performance Comparison (mAP, %) on the Oxford 17-category Flowers dataset.

Method	Accuracy (%)
Gehler and Nowozin [15]	84.40
Fernando *et al*. [14]	94.50
Yang *et al*. [38]	97.80
CNN-SVM (w/o fine-tuning)	91.52
CNN ft (1 img/cls)	87.25
CNN ft (1 img/cls + 20 rcls/img)	89.02
CNN ft (5 imgs/cls)	92.10
CNN ft (5 imgs/cls + 20 rcls/img)	93.28
CNN ft (40 imgs/cls)	95.23
CNN ft (40 imgs/cls + 20 rcls/img)	96.84

∗ cls, img, rcl denote class, image, and recall, respectively.

Classification Performance v.s. Different Sample Sizes Used in Fine-tuning Figure 6 shows the classification accuracy with respect to different sizes of training samples used in fine-tuning the network. We observe that (1) increasing the size of the training samples can improve the performance of the fine-tuned model, and (2) including the recalled images in the training samples can further improve the performance. These observations are consistent with those in the 17 category dataset, showing that enriching training data is effective for fine-tuning a deep CNN with few training examples.

Performance Comparison to Other Methods Table 2 lists the performance of other approaches and ours. Both



∗ cls, img, and rcl denote class, image, and recall, respectively.

Figure 6: Comparison of the classification accuracy on the Oxford 102 category Flower dataset with respect to different numbers of images used in the fine-tuning process. Higher classification accuracy can be achieved when more training samples are available. This observation is consistent with that in the smaller 17 category dataset.

Table 2: Performance Comparison (mAP, %) on the Oxford 102-category Flowers dataset.

Method	Accuracy (%)
Cai *et al*. [6]	80.00
Angelova and Zhu [1]	80.66
CNN-SVM (w/o fine-tuning) [32]	74.70
CNNaug-SVM (w/o fine-tuning) [32]	86.80
CNN ft (1 img/cls)	82.41
CNN ft (1 img/cls + 5 rcls/qry)	82.66
CNN ft (5 imgs/cls)	88.37
CNN ft (5 imgs/cls + 5 rcls/img)	89.36
CNN ft (10 imgs/cls)	89.00
CNN ft (10 imgs/cls + 5 rcls/img)	90.85

∗ cls, img, rcl denote class, image, and recall, respectively.

the CNN-SVM and CNNaug-SVM use the *OverFeat* [33] features for training SVM classifiers. The main difference is that the CNNaug-SVM applies data augmentation step and uses 16 representations for each sample (original image, 5 crops, 2 rotation and their mirrors). Our method (only 1 image per class in fine-tuning) achieves an accuracy of 82.41%, performing favorably against most approaches. With fine-tuning the network on 5 images per class, our method achieves an accuracy of 88.37% and outperforms the other state-of-the-art methods (86.80%). Our method can achieve a best performance of 90.85% for the 102-classes-of-flowers problem when enriching the per-class training images to 10 with the recalled images.

4.4. Effects of Fine-Tuning on Filters

We visualize the outputs of each layer in the pre-trained model and the fine-tuned network to investigate the differences in response to a same input. As shown in Figure 7,

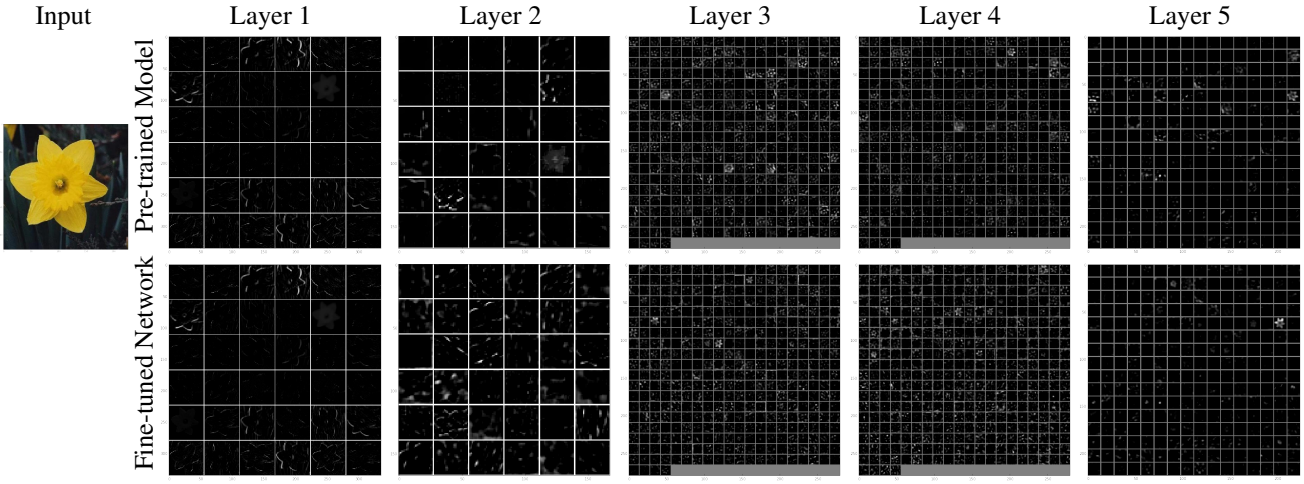


Figure 7: Visualization of each layer's outputs of the pre-trained model and the fine-tuned network for a given input. The pre-trained model is the one proposed by Krizhevsky *et al*. [20]. The fine-tuned network is obtained by using 40 images per class and 20 recalled images per query. The layer outputs of two networks are visually different: subtle differences in Layer 1 and more distinct differences in the subsequent layers. While the shallower layers (layer 2 to 4) of the fine-tuned one reflect more component-level information such as petal and stamen, the deeper layer (layer 5) becomes more discriminative and tends to capture higher level semantic information for classification.

the outputs of a same layer in two networks are visually different. The differences are subtle in layer 1, but become more distinct in the subsequent layers. This indicates the fine-tuning process may have a stronger effect on the neuron connections in the subsequent layers than it does on the first few layers. The fine-tuned network becomes more specific to the flower recognition task and thus exhibits greater activation to a flower input than the pre-trained model. As can be seen, while the shallower layers (layer 2 to 4) of the fine-tuned one reflect more component-level information such as petal and stamen, the deeper layer (layer 5) becomes more discriminative and tends to capture higher level semantic information for classification.

Although the fine-tuned network is designed specifically for the flower recognition, it may fail to distinguish the flowers whose shapes and colors are significantly similar. Figure 8 shows the correctly classified and misclassified images. The issue of misclassification may be caused due to insufficient amount of training samples in fine-tuning.

4.5. Computational Complexity

Recalling similar patterns from the pre-trained network takes around 3 minutes per query when all the images in ImageNet are included in the search. Because our goal is to retrieve highly relevant samples, taking into account only the images with the labels under "plant" category and discarding those irrelevant ones is a natural way to speed up the search. With only around 50,000 images in the search pool, the recall process takes a few seconds per query. In fine-tuning, the network became stable after 100 and 300 iterations for the 17-category and 102-category datasets, respectively. This process takes roughly 10 to 20 minutes. Classification takes around 0.07 seconds per image. All the tests are performed on a machine with Geforce GTX Titan Black GPU.

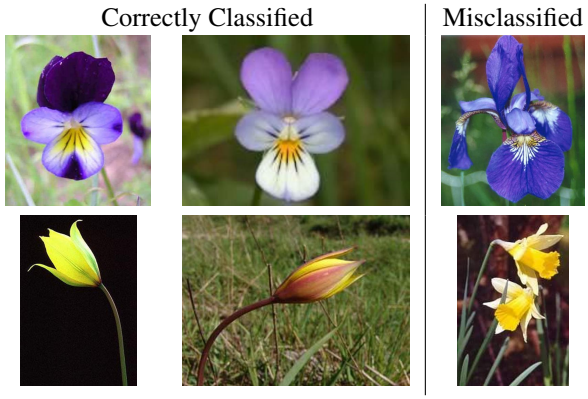


Figure 8: Correctly classified test images and misclassified ones. The top row shows images classified as Pansy; the first two are correctly classified but the correct category of the third is Iris. The bottom row shows images classified as Tulip; the third is misclassified, which belongs to Daffodil.

5. Conclusions

Pre-training a deep CNN on large-scale datasets to learn rich image representations and fine-tuning the network in the new domain has been an effective way in many recognition tasks. We present a framework to address the problem of fine-tuning a deep CNN with few training samples. In particular, we propose to enrich the training data by recalling similar patterns from the pre-trained model. Both the

images in the new domain and the recalled images form a new set of training samples in the fine-tuning process. Our results on the Oxford flower datasets show that enriching the training data can help the network learn discriminant image descriptors in the new domain when the training data is scarce. In future, we plan to include human in the loop of recall process for determining the optimal set of recalled images. With human intervention, we can guarantee better quality of recalled images for further semi-supervision.

References

[1] A. Angelova and S. Zhu. Efficient object detection and segmentation for fine-grained recognition. In *Proc. CVPR*, pages 811–818, 2013.

[2] D. P. Ausubel. *The Psychology of Meaningful Verbal Learning*. New York: Grune and Stratton, 1963.

[3] E. Bart and S. Ullman. Cross-generalization: Learning novel classes from a single example by feature replacement. In *Proc. CVPR*, volume 1, pages 672–679, 2005.

[4] I. Biederman. Recognition-by-components: a theory of human image understanding. *Psychological review*, 94(2):115, 1987.

[5] C. Carpineto and G. Romano. A survey of automatic query expansion in information retrieval. *ACM Comput. Surv.*, 44(1):1:1–1:50, 2012.

[6] Y. Chai, V. Lempitsky, and A. Zisserman. BiCoS: A bi-level co-segmentation method for image classification. In *Proc. ICCV*, 2011.

[7] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. In *Proc. BMVC*, 2014.

[8] D. C. Ciresan, U. Meier, and J. Schmidhuber. Multi-column deep neural networks for image classification. In *Proc. CVPR*, pages 3642–3649, 2012.

[9] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A large-scale hierarchical image database. In *Proc. CVPR*, pages 248–255, 2009.

[10] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. DeCAF: A deep convolutional activation feature for generic visual recognition. In *Proc. ICML*, pages 647–655, 2014.

[11] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The Pascal Visual Object Classes (VOC) challenge. *Int'l J. Computer Vision*, 88(2):303–338, 2010.

[12] L. Fei-Fei, R. Fergus, and P. Perona. One-shot learning of object categories. *Trans. PAMI*, 28(4):594–611, 2006.

[13] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer Vision and Image Understanding*, 106(1):59–70, 2007.

[14] B. Fernando, E. Fromont, and T. Tuytelaars. Effective use of frequent itemset mining for image classification. In *Proc. ECCV*, pages 214–227, 2012.

[15] P. Gehler and S. Nowozin. On feature combination for multiclass object classification. In *Proc. ICCV*, pages 221–228, 2009.

[16] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proc. CVPR*, pages 580–587, 2014.

[17] M. Guillaumin, J. Verbeek, and C. Schmid. Multimodal semi-supervised learning for image classification. In *Proc. CVPR*, pages 902–909. IEEE, 2010.

[18] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.

[19] T. Joachims. Transductive inference for text classification using support vector machines. In *Proc. ICML*, volume 99, pages 200–209, 1999.

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *Proc. NIPS*, pages 1106–1114, 2012.

[21] V. Lavrenko and W. B. Croft. Relevance based language models. In *Proc. Annual Int'l ACM SIGIR Conf. Research and Development in Information Retrieval*, pages 120–127, 2001.

[22] Q. V. Le, M. Ranzato, R. Monga, M. Devin, K. Chen, G. Corrado, J. Dean, and A. Y. Ng. Building high-level features using large scale unsupervised learning. In *Proc. ICML*, 2012.

[23] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[24] K. S. Lee and W. B. Croft. A deterministic resampling method using overlapping document clusters for pseudo-relevance feedback. *Information Processing and Management*, 49(4):792–806, 2013.

[25] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int'l J. Computer Vision*, 60(2):91–110, 2004.

[26] G. Mesnil, Y. Dauphin, X. Glorot, S. Rifai, Y. Bengio, I. J. Goodfellow, E. Lavoie, X. Muller, G. Desjardins, D. Warde-Farley, P. Vincent, A. Courville, and J. Bergstra. Unsupervised and transfer learning challenge: a deep learning approach. *J. Machine Learning Research-Proceedings Track*, 27:97–110, 2012.

[27] E. G. Miller, N. E. Matsakis, and P. A. Viola. Learning from one example through shared densities on transforms. In *Proc. CVPR*, volume 1, pages 464–471, 2000.

[28] M.-E. Nilsback and A. Zisserman. A visual vocabulary for flower classification. In *Proc. CVPR*, pages 1447–1454, 2006.

[29] M.-E. Nilsback and A. Zisserman. Automated flower classification over a large number of classes. In *Proc. Indian Conf. Computer Vision, Graphics and Image Processing*, 2008.

[30] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *Proc. CVPR*, pages 1717–1724, 2014.

[31] F. Perronnin, J. Sánchez, and T. Mensink. Improving the Fisher kernel for large-scale image classification. In *Proc. ECCV*, pages 143–156, 2010.

[32] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson. CNN features off-the-shelf: An astounding baseline for recognition. In *Proc. CVPR Workshops*, 2014.

[33] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. In *Proc. ICLR*, 2014.

[34] P. Sermanet, K. Kavukcuoglu, S. Chintala, and Y. LeCun. Pedestrian detection with unsupervised multi-stage feature learning. In *Proc. CVPR*, pages 3626–3633, 2013.

[35] S.-Y. Wang, W.-S. Liao, L.-C. Hsieh, Y.-Y. Chen, and W. H. Hsu. Learning by expansion: Exploiting social media for image classification with few training examples. *Neurocomputing*, 95:117–125, 2012.

[36] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(2):210–227, 2009.

[37] N. Xie, H. Ling, W. Hu, and X. Zhang. Use bin-ratio information for category and scene classification. In *Proc. CVPR*, pages 2313–2319, 2010.

[38] M. Yang, L. Zhang, X. Feng, and D. Zhang. Sparse representation based fisher discrimination dictionary learning for image classification. *Int'l J. Computer Vision*, 109(3):209–232, 2014.

[39] G. Ye, D. Liu, I.-H. Jhuo, and S.-F. Chang. Robust late fusion with rank minimization. In *Proc. CVPR*, pages 3021–3028, 2012.

[40] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *Proc. ECCV*, pages 818–833, 2014.

[41] Z.-H. Zhou, D.-C. Zhan, and Q. Yang. Semi-supervised learning with very few labeled training examples. In *Proc. AAAI*, volume 22, page 675, 2007.