In Defense of Fully Connected Layers in Visual Representation Transfer*

Chen-Lin Zhang, Jian-Hao Luo, Xiu-Shen Wei, Jianxin Wu

National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, China
{zhangcl, luojh, weixs, wujx}@lamda.nju.edu.cn

Abstract Pre-trained convolutional neural network (CNN) models have been widely applied in many computer vision tasks, especially in transfer learning tasks. In transfer learning, the target domain may be in a different feature space or follow a different data distribution, compared to the source domain. In CNN transfer tasks, we often transfer visual representations from a source domain (e.g., ImageNet) to target domains with fewer training images or have different image properties. It is natural to explore which CNN model performs better in visual representation transfer. Through visualization analyses and extensive experiments, we show that when either image properties or task objective in the target domain is far away from those in the source domain, having the fully connected layers in the source domain pre-trained model is essential in achieving high accuracy after transferring to the target domain.

Keywords: Deep Learning, Computer Vision, Fully Connected Layers

1 Introduction

Convolutional neural network (CNN), which is now pervasive in computer vision [7], is a very successful visual representation learning approach. Research of CNN in artificial intelligence includes not only the real-world applications, but also the fundamental developments of CNN itself. However, a systematic study of classic CNN modules (e.g., the fully connected layer) in various setup (i.e., different from the conventional classification usage) is missing in the literature.

The fully connected (FC) layer is one of the most fundamental modules in CNN. It is widely used in traditional CNN models [7,14]. However, it is known that FC may cause overfitting, and it requires millions of parameters [14]. In recent CNN models (such as GoogLeNet [15] and ResNet [5]), a global average pooling layer replaces the last FC layers, which has much fewer parameters and improves the classification accuracy on the challenging ImageNet [12] dataset. Thus, more and more deep models prefer discarding FC for better performance and efficiency [5,8,9,18]. The utilities of FC layers in CNN have declined in recent research.

In this paper, however, we are in defense of FC layers in visual representation transfer. In visual representation transfer, the popular way to transfer is fine-tuning, which uses a pre-trained model from the source domain to initialize the deep model in the

* This work was supported by the Collaborative Innovation Center of Novel Software Technology and Industrialization. J. Wu is the corresponding author.

target domain, and updates its parameters using the target domain data. In general, we have various image data and pre-trained models in source domains like the popular ImageNet [12] data. While, in target domains, there will be fewer image data than ImageNet, or even with different kinds of images.

In our experiments, we treat ImageNet as the source domain, and take diverse kinds of image data as the target domains to perform visual representation transfer, i.e., different numbers of images, object-centric/scene-centric properties and even different granularities (cf. Table 1). We show that when the target domain has a small number of images or the difference between source and target domains is large, FC layers play an important role in achieving high accuracy in target domains by fine-tuning the pre-trained model. To the best of our knowledge, this is the first empirical study that shows the importance of FC layers in transferring CNN visual representations.

2 Related Work

CNN has become the *de facto* standard in many computer vision research. Alex-Net [7] used a model with five convolution layers (conv_1, conv_2, ..., conv_5) with three FC layers (fc_6, fc_7 and fc_8), and achieved rank 1^{st} in the ILSVRC12 classification task. They can both add non-linearity to the models and finish the classification. In spite of these functions, FC layers have serious disadvantages: easily getting overfitted, hard to converge during training, and hampering the generalization ability [8].

Researchers have proposed the global average pooling strategy to replace the FC layers [8]. The global average pooling layer has no parameter, which summarizes the spatial information using an average, and can be seen as a regularizer. GoogLeNet [15] and ResNet [5] are two typical examples without the last FC layers. Both GoogLeNet and ResNet used the global average pooling layer to replace the last fully connected layers, and have achieved best results in the ImageNet competition in 2014 and 2015, respectively.

On the other hand, VGG-Nets [14] also achieved state-of-the-art results. VGG-Nets have similar architecture as that of the Alex-Net. The FC layers are still used in VGG-Nets. An interesting observation is that the VGG-Nets have become a popular feature extraction tool in various computer vision tasks. GoogLeNet and ResNet, however, perform well in tasks with large datasets or tasks which are similar to source domains, but they are not very popular in these transfer learning tasks, and even fail in some tasks [10,4]. In this paper, we show that it is because the FC layers in VGG-16 leads to high accuracy in visual representation transfer.

There is some research on deep transfer learning. It has been shown that pre-trained CNN descriptors are very powerful in many tasks [13,2]. Performing transfer learning based on pre-trained models will benefit the system's accuracy [21]. This procedure is called fine-tuning, and it is also showed that fine-tuning all layers will also improve the final result [20].

In practice, fine-tuning starts from an already learned model like VGG-16, then slightly modifies the network structure. Fine-tuning then initializes the network weights using pre-trained model, and starts training on the target dataset. It has become the most popular transfer learning method in many computer vision tasks in the deep learning

scenario. In this paper, we will mainly focus on this type of visual representation transfer learning.

3 In Defense of FC in CNN Transfer Learning

As aforementioned, FC layers are shown as inefficient and inefficacious when training CNN from scratch in only the source domain. In this section, however, we show that the FC layers are essential when transferring the representation in CNN, especially when the source and target domains are far away from each other. To support such a statement, we conduct visual representation transfer experiments on the VGG [14] and ResNet [5] architectures. Based on the two architectures, CNN models with and without FC layers are employed for ablation experiments. Moreover, we prepare several computer vision tasks in different target domains. Finally, by comparing the performance of these CNN models on these target domains, we establish the importance of FC layers in visual representation transfer.

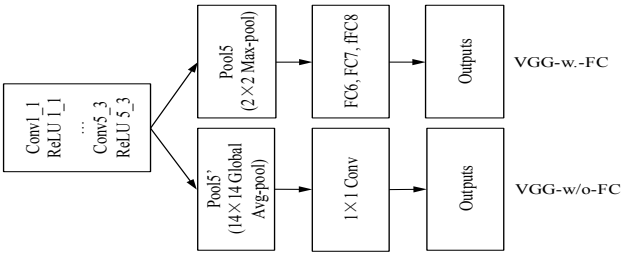
3.1 CNN Models with and without FC Layers

Experiments are conducted on different CNN models with and without FC based on two architectures. Thus, we have four CNN models from the source domain, i.e., the ImageNet [12] data. Two of them are based on VGG-16 [14], and the other two are based on ResNet [5].

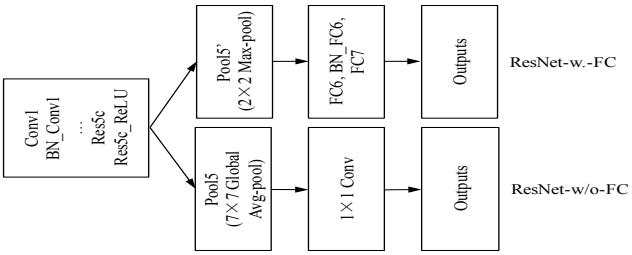
For VGG-16, we name `VGG-w.-FC` to indicate the pre-trained VGG-16 model with the FC layers. In addition, in order to obtain a VGG model without FC from the source domain, we replace the pool_5 layer with a global average pooling layer and remove all subsequent FC layers in original VGG-16. Then, a $1 \times 1 \times 1000$ convolution layer is added to output the predicted results. The modified VGG-16 model is named as `VGG-w/o-FC`. Then, the ImageNet dataset is used to fine-tune `VGG-w/o-FC` until converging. The structures of `VGG-w.-FC` and `VGG-w/o-FC` are illustrated in Fig. 1a.

A few interesting observations are obtained on these two VGG-based models. First, it can be considered as a transfer learning task in which the target domain is the same as the source domain. For our `VGG-w/o-FC`, we achieved 10.59% Top-5 error on the ILSVRC 2012 validation set, which is 0.95% lower than that of `VGG-w.-FC`. The size (i.e., number of parameters) of `VGG-w/o-FC` is only 11.00% of that of `VGG-w.-FC` (15.22 vs. 138.34 million), but it has higher accuracy than that of `VGG-w.-FC`. This comparison corroborates the fact that removing FC layers is advantageous when training CNN from scratch with enough training images.

For ResNet models, we use the public ResNet-50 model pre-trained on ImageNet as `ResNet-w/o-FC` model. Because `ResNet-w/o-FC` has no FC layer, we add a 2×2 local max pooling layer after removing the global average pooling layer and final FC layer. Then, we add a 1024-d FC layer, followed by a batch normalization layer, and finally we add a 1000-d FC layer for classification. We name this model as `ResNet-w.-FC`. ResNet-w/o-FC got 7.82% Top-5 error on the ImageNet validation set, and ResNet-w.-FC got 8.64% Top-5 error, which also indicates that removing FC layers is advantageous when training CNN from scratch with enough training images.



(a) VGG architecture.



(b) ResNet architecture.

Figure 1: Network structures of different CNN architectures.

When both models are transferred to other target domains, we change the number of nodes in the last layer according to the target domain, using the same initialization method to set the new values for the last layer, and fine-tune all parameters in the models using the training data in target domains.

In addition, we also conduct experiments on two more VGG-based models during transferring, i.e., `VGG-w.-FC-fix` and `VGG-w/o-FC-fix`. For these models, we fix the representation learning parts. In other words, the CNN representations are used unchangeably.

3.2 Image Data in Different Target Domains

We use four datasets (target domains, see Table 1 for a summary of their properties) and perform three image tasks (classification, content-based retrieval and localization) to study the performance differences with or without the FC layers.

First, four image classification datasets are included (Caltech-101 [3], Indoor-67 [11], RGB-NIR scene [1] and CUB200-2011 [16]). The tasks of these datasets are the same as the baseline models (i.e., classification). However, the images differ from ImageNet in their properties. Second, we transfer the baseline models to perform fine-grained unsupervised image retrieval and object localization, in which the tasks are significantly different in the source and target domains. Examples of these datasets are shown in Figure 2. Details of these datasets are as follows.

(a) ImageNet

(b) Caltech-101 (c) Indoor-67 (d) 9-class RGB (e) 9-class NIR (f) CUB

Figure 2: Example images from the source domain (ImageNet) and target domains, i.e., (b)-(f). We organize these datasets such that the similarity (including *object types*, *image types*, *imaging sensors* and *category granularities*) between the source and target domain decreases from left to right. Note that image classification is performed in the source (ImageNet) and (b)-(f), and fine-grained retrieval and object localization is performed on (f).

- **Caltech-101.** It has 101 categories of objects [3]. This dataset is the most similar one to ImageNet, because the categories in Caltech-101 are mostly included in the categories in ImageNet.
- **MIT Indoor-67.** It has 67 indoor scene categories [11]. This dataset is less similar to the source ImageNet dataset than Caltech-101. Instead of recognizing objects, the categories are characterized by *scenes*. However, some scene categories are common in both MIT indoor and ImageNet.
- **9-class RGB-NIR scene.** This scene recognition dataset is proposed in [1]. It includes nine scene categories in the RGB-NIR image format. In our study, it is divided into two parts: the RGB image and the NIR image. It has nine classes of outdoor scenes.
  The RGB images in this dataset is getting more distant to ImageNet, because the outdoor scenes in this dataset is different from those in ImageNet. The NIR part of this dataset is taken by different sensors than the ImageNet (NIR vs. RGB). Due to this difference in imaging sensor, the NIR images are dissimilar to the ImageNet images.
- **CUB200-2011** [16] contains 11788 images of 200 fine-grained bird species. We perform both classification and unsupervised localization/retrieval tasks on CUB200-2011. Fine-grained image classification is performed as similar as the other datasets. For unsupervised fine-grained image retrieval and object localization, we use the SCDA method [17] to show performance of different CNN models.

3.3 Visualization and Observations

Before comparing numerical accuracy rates in various target domains, we take `VGG-w.-FC` and `VGG-w/o-FC` as examples to visualize for giving us some intuitions about their differences. Figure 3 shows their corresponding visualization.

We visualize both forward and backward feature maps. For each forward feature map, we show the results after fine-tuning. For the backward feature maps, we visualize

Table 1: Summary of the datasets' properties.

Dataset	♯ images	♯ classes	Style	Granularity	Color space
Caltech-101	9,145	102	Object	Generic	RGB
Indoor-67	6,700	67	Scene	Generic	RGB
9class RGB	477	9	Scene	Generic	RGB
9class NIR	477	9	Scene	Generic	NIR
CUB200-2011	11,788	200	Object	Fine-grained	RGB

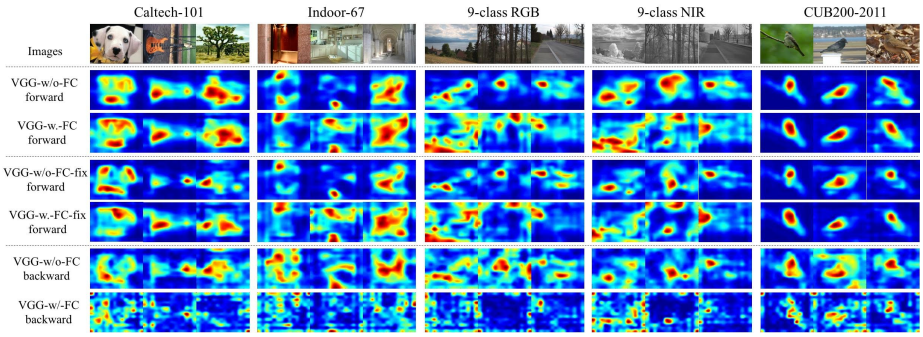


Figure 3: Visualization of the activations (of relu_{5_3}) for the four baseline models on four classification datasets. The first row shows the input images. The second, third, fourth and fifth rows show the forward feature maps of `VGG-w/o-FC`, `VGG-w.-FC`, `VGG-w/o-FC-fix`, `VGG-w.-FC-fix`, respectively. The sixth and seventh rows show the backward feature maps of `VGG-w/o-FC` and `VGG-w.-FC` models, respectively. Note that we organize the figures such that every two rows between the horizontal bars are directly comparable, i.e., they differ only by the existence or missing of the fully connected layers. Color code is used to visualize the values: the red regions mean larger values and blue regions refer to smaller values.

the gradient when the two non-fix models start to fine-tune on the visualized input image. For each visualization, we sum the response values of all the channels in the relu_{5_3} layer for each deep network.

One obvious difference between models with and without FC layers is: while models without FC layers has its activation map (i.e., the red regions) concentrated around the center object, those with the FC layers (e.g., `VGG-w.-FC`) has activation maps that is more *distributed*, i.e., the activations scatter in many locations in the image.

Hence, we conjecture that when FC layers are missing, the activations is *too concentrated around the object* (i.e., *only features tightly related to the source domain has strong responses*). This close relationship makes such models both efficient and effective in the source domain, but at the same time may make it inappropriate to transfer to a target domain if the source and target are distant from each other. Those models with FC layers show a different property. Although they might be less effective in the source domain, its distributed activations *enable them to capture useful image features in target domains*, even if the target is dissimilar to the source domain.

This difference can be partly explained by the different pooling action after relu_{5_3} in these two models (cf. Figure 1). In `VGG-w/o-FC`, the layer conv_{5_3} is updated directly using the error signal in the classification. Hence, the visual representation is

Table 2: Comparison of classification accuracy on four datasets. The best result in a column of each sub-table is marked in bold. Note that, the "-fix" version of VGG models indicates their representation learning parts are fixed.

	FC	Caltech-101	indoor-67	RGB scene	NIR scene	CUB
VGG-w.-FC	✓	87.24%	**66.27%**	**80.20%**	**76.40%**	**73.24%**
VGG-w/o-FC	✗	**88.17%**	64.97%	78.80%	75.56%	71.90%
VGG-w.-FC-fix	✓	88.64%	**66.56%**	**81.60%**	**79.12%**	**68.42%**
VGG-w/o-FC-fix	✗	**89.40%**	64.86%	77.76%	76.52%	67.90%
ResNet-w.-FC	✓	90.89%	**74.75%**	**90.20%**	**87.87%**	**81.81%**
ResNet-w/o-FC	✗	**91.03%**	74.44%	89.90%	86.86%	81.50%

highly focused for classification of the source domain. In VGG-w.-FC, the error signal will first affect the fc_8 layers, then fc_7 and fc_6, finally it will affect $conv_{5\_3}$. The FC layers fc_6 and fc_7 act like "firewalls" in the transfer process such that the features in $conv_{5\_3}$ are not directly affected by the classification error. Hence it will reflect more general image structures.

As the example in the first column of Figure 3 shows, VGG-w/o-FC only focuses on the dog's mouth or the ear. Because the target is similar to the source domain in this case, this concentration of attention is a desired property. We observe similar concentration in the examples from fourth column to sixth column. However, when we move on the more distant target domains, the models without FC layers only concentrate around few small regions, which fails to capture useful information in the target domain. VGG-w.-FC, which has the FC layers, on the contrary, activates on many regions that are useful in describing the target domain image. For example, it activates on different types of objects in the fifth column (for indoor images) and different trees in the tenth column. For backward feature maps, we can easily see the same findings: VGG-w/o-FC mainly focuses on parts of dogs, while VGG-w.-FC is more distributed, both for dog parts and background parts.

4 Experiments

We first describe the experimental setting, then the results and analyses follow.

4.1 Experimental Setup

For all the classification tasks, the images are resized to 224×224, and we did not use additional data augmentation techniques. In validation, we use the one image policy. For the source domain models, VGG-w/o-FC is trained by the Caffe toolbox [6] with learning rate 10^{-3}, weight decay 5×10^{-4} and momentum 0.9. For training ResNet-w.-FC, we use the Torch toolkit with the same hyper-parameter values to train ResNet-w.-FC.

Regarding the dataset settings, for most datasets, we follow the traditional protocols or original training/test splitting provided by the datasets.

Specifically, on Caltech-101, We randomly sample 30 images per category for training, and the remaining up to 50 images per category for testing . We repeat five random splits and report the average accuracy. For the 9-class RGB-NIR dataset, we use

Table 3: Comparison of different outputs' SCDA object localization accuracy on CUB200-2011 with different IoU.

Models	FC	0.35	0.4	0.45	0.5	0.55	0.6	0.65	0.7
VGG-w.-FC 7×7	✓	**91.82%**	**87.71%**	**82.86%**	**76.79%**	**68.95%**	**59.89%**	**48.71%**	**36.35%**
VGG-w/o-FC 7×7	✗	75.94%	69.40%	62.00%	54.88%	47.29%	40.14%	32.78%	25.72%
VGG-w.-FC 14×14	✓	**92.35%**	**88.97%**	**84.28%**	**79.46%**	**73.52%**	**65.86%**	**56.87%**	**46.63%**
VGG-w/o-FC 14×14	✗	77.79%	71.07%	64.20%	56.42%	49.72%	43.30%	36.04%	28.98%

Table 4: Comparison of different outputs' SCDA image retrieval accuracy on CUB200-2011.

Models	FC	Avg. pooling		Max pooling		Avg.+Max pooling	
		Top-1	Top-5	Top-1	Top-5	Top-1	Top-5
VGG-w.-FC 7×7	✓	**56.42%**	**63.14%**	**58.35%**	**64.18%**	**59.72%**	**65.79%**
VGG-w/o-FC 7×7	✗	22.26%	29.33%	24.44%	31.51%	26.20%	33.31%
VGG-w.-FC 14×14	✓	**55.33%**	**62.04%**	**58.03%**	**63.93%**	**59.08%**	**65.45%**
VGG-w/o-FC 14×14	✗	22.51%	30.06%	24.21%	31.48%	26.61%	33.91%

the RGB images and NIR images for two independent classification tasks. The NIR 1-channel images are replicated into 3-channels as model inputs. For both RGB and NIR tasks, we follow the setup in [19]: 42 images are randomly chosen for training per category, and the rest 11 images for testing . The averaged accuracy is reported on five random splits. For MIT Indoor-67 and CUB200-2011, we follow the training and test splitting included with these datasets.

For fine-tuning models in target domains, we set the base learning rate to 10^{-5}, momentum to 0.9 and weight decay to 5×10^{-4}. The parameters of the last layer are initialized from a zero-mean Gaussian with the standard deviation as 10^{-3}.

For fine-grained image retrieval on CUB200-2011, [17] proposed a simple but effective method to do unsupervised fine-grained image retrieval and object localization, i.e., Selective Convolutional Descriptor Aggregation (SCDA). SCDA only needs a model pre-trained from ImageNet, hence we choose it as our experimental method. SCDA utilizes layers' activations and descriptor aggregation such as average- and max-pooling. For VGG-w.-FC, the activations of relu_{5_3} and pool_5 are used, which are a $14 \times 14 \times 512$ tensor (VGG-w.-FC 14×14 in Table 3 and 4) and a $7 \times 7 \times 512$ tensor (VGG-w.-FC 7×7 in Table 3 and 4), respectively. As the comparisons with VGG-w/o-FC, we first remove the layers after relu_{5_3}, then add a 2×2 max pooling layer (also call it pool_5). Similarly, the outputs of relu_{5_3} and pool_5 are extracted, which also have $14 \times 14 \times 512$ and $7 \times 7 \times 512$ activations, respectively. They are shown as VGG-w/o-FC 14×14 and VGG-w/o-FC 7×7 in Table 3 and 4. Now we have four methods to compare in total.

In Table 3, we report the object localization accuracy on CUB200-2011 with different Intersection-over-Union (IoU) ratios. In Table 4, the Top-1 and Top-5 mAP as the retrieval performance are reported for these four models. Additionally, since SCDA did not work well on the ResNet-50 based models, the localization accuracy and retrieval results of these models are not reported (about 20% lower than VGG models).

4.2 Results and Analyses

Results of image classification tasks are listed in Table 2. It is obvious that the best result for each dataset appear mostly in `w.-FC` models except the Caltech-101 dataset. That is, as the target domain is getting more dissimilar to the source domain, having the fully connected layers are becoming more important. However, when the source and target domains are similar, the `w/o-FC` models are more accurate. In Caltech-101, `VGG-w/o-FC` outperforms `VGG-w.-FC` by 0.93%. While in dissimilar datasets like CUB200-2011 and 9-class RGB-NIR, `VGG-w.-FC` leads `VGG-w/o-FC` by a 1.34%, 0.84% and 1.40% margin. For ResNet-50 based models, they show the same conclusion as VGG based models: `ResNet-w.-FC` leads `ResNet-w/o-FC` by 0.5% to 1% in all four datasets except the Caltech-101 dataset.

The `-fix` versions of CNN models fix the representation learning parts, whose results are shown in Table 2. In transferring to dissimilar target domains, the models with FC layers consistently outperform those without FC layers; and `VGG-w.-FC-fix` even has a significant improvement over `VGG-w.-FC`, the non-fixed version. We guess that for target domains with small data, only fine-tune the FC layers can prevent models from overfitting, and achieve higher accuracy. We conjecture that when the target domain is distant from the source domain, and when the number of training images is very small, applying the visual representation in a pre-trained model (learned with the FC layers) is the optimal option.

For fine-grained image retrieval and object localization tasks, because SCDA is an unsupervised method, the performance is directly decided by the pre-trained model itself. In these two tasks, `VGG-w.-FC` based models performs significantly better than `VGG-w/o-FC` based models in all situations. In image retrieval, `VGG-w.-FC` based models get about 58% Top-1 and 63% Top-5 accuracy, but `VGG-w/o-FC` based models can only get about 25% Top-1 and 30% Top-5 accuracy, leaving a 30% gap. Similar gaps exist in the object localization tasks. These two tasks are totally different from the ImageNet classification task, which may explain this gap: when the source and target domains differ not only in image properties but also required tasks, having the FC layers are essential too.

Hence, when the source and target domains are similar in both image properties and task objectives, we recommend not using any FC layer in the source domain. However, we defend the importance of FC layers if there is significant dissimilarity in either image property or task objective.

5 Conclusion and Future Work

In this paper, we have studied the usage of fully connected layers in visual representation transfer. By performing visualization analyses and experiments on classification, fine-grained retrieval and object localization on various kinds of datasets in target domains, we conclude that when the target domain is not far away from the source domain, fully connected layers can be replaced by global average pooling for better efficiency and accuracy. However, when a large difference exists in either image property or task objective, fully connected layers are essential in visual representation transfer.

In the future, we will try to improve the performance of CNN models with global average pooling. We want to maintain its small model size and efficiency, and make it suitable for visual representation transfer to distant target domains.

References

1. Brown, M., Süsstrunk, S.: Multi-spectral SIFT for scene category recognition. In: CVPR. pp. 177–184 (2011)
2. Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., Darrell, T.: DeCAF: A deep convolutional activation feature for generic visual recognition. In: ICML. pp. 647–655 (2014)
3. Fei-Fei, L., Fergus, R., Perona, P.: Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. CVIU 106, 59–70 (2007)
4. Girshick, R.: Fast R-CNN. In: ICCV. pp. 1440–1448 (2015)
5. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. pp. 770–778 (2016)
6. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture for fast feature embedding. In: ACM MM. pp. 675–678 (2014)
7. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: NIPS. pp. 1097–1105 (2012)
8. Lin, M., Chen, Q., Yan, S.: Network in network. In: ICLR. pp. 1–10 (2014)
9. Lin, T.Y., RoyChowdhury, A., Majiu, S.: Bilinear cnn models for fine-grained visual recognition. In: ICCV. pp. 1449–1457 (2015)
10. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: CVPR. pp. 3431–3440 (2015)
11. Quattoni, A., Torralba, A.: Recognizing indoor scenes. In: CVPR. pp. 413–420 (2009)
12. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet large scale visual recognition challenge. IJCV 115, 211–252 (2015)
13. Sharif Razavian, A., Azizpour, H., Sullivan, J., Carlsson, S.: CNN features off-the-shelf: An astounding baseline for recognition. In: CVPR 14 Workshops. pp. 806–813 (2014)
14. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: ICLR. pp. 1–14 (2015)
15. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: CVPR. pp. 1–9 (2015)
16. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The Caltech-UCSD birds-200-2011 dataset. Tech. Rep. CNS-TR-2011-001, California Institute of Technology (2011)
17. Wei, X.S., Luo, J.H., Wu, J., Zhou, Z.H.: Selective convolutional descriptor aggregation for fine-grained image retrieval. IEEE TIP 26(6), 2868–2881 (2017)
18. Wei, X.S., Xie, C.W., Wu, J.: Mask-cnn: Localizing parts and selecting descriptors for fine-grained image recognition. arXiv preprint arXiv:1605.06878 (2016)
19. Xiao, Y., Wu, J., Yuan, J.: mCENTRIST: A multi-channel feature generation mechanism for scene categorization. IEEE TIP 23, 823–836 (2014)
20. Yosinski, J., Clune, J., Bengio, Y., Lipson, H.: How transferable are features in deep neural networks? In: NIPS. pp. 3320–3328 (2014)
21. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: ECCV. pp. 818–833. Springer (2014)