

Multi-branch and Multi-scale Attention Learning for Fine-Grained Visual Categorization

Fan Zhang^[0000-0003-2504-4651], Meng Li^[0000-0002-1543-5029], Guisheng Zhai^[0000-0001-9901-010X], and Yizhao Liu^[0000-0002-9549-7317]

School of MEIE , China University of Mining and Technology (Beijing), Beijing, China

sqt1800407119@student.cumtb.edu.cn

Abstract. ImageNet Large Scale Visual Recognition Challenge (ILSVRC) is one of the most authoritative academic competitions in the field of Computer Vision (CV) in recent years. But applying ILSVRC's annual champion directly to fine-grained visual categorization (FGVC) tasks does not achieve good performance. To FGVC tasks, the small inter-class variations and the large intra-class variations make it a challenging problem. Our attention object location module (AOLM) can predict the position of the object and attention part proposal module (APPM) can propose informative part regions without the need of bounding-box or part annotations. The obtained object images not only contain almost the entire structure of the object, but also contains more details, part images have many different scales and more fine-grained features, and the raw images contain the complete object. The three kinds of training images are supervised by our multi-branch network. Therefore, our multi-branch and multi-scale learning network(MMAL-Net) has good classification ability and robustness for images of different scales. Our approach can be trained end-to-end, while provides short inference time. Through the comprehensive experiments demonstrate that our approach can achieves state-of-the-art results on CUB-200-2011, FGVC-Aircraft and Stanford Cars datasets. Our code will be available at <https://github.com/ZF1044404254/MMAL-Net>

Keywords: FGVC, Classification, Attention, Location, Scale.

1 Introduction

How to tell a dog's breed? This is a frequently asked question because dogs have similar characteristics. The FGVC direction of CV research focuses on such issues, and it is also called sub-category recognition. In recent years, it is a very popular research topic in CV, pattern recognition and other fields. It's purpose is to make a more detailed sub-class division for coarse-grained large categories (e.g. classifying bird species [1], aircraft models [2], car models [3], etc.).

Many papers [4,5,6,7] have shown that the key to fine-grained visual categorization tasks lies in developing effective methods to accurately identify informative regions in an image. They leverage the extra annotations of bounding box and part annotations to localize significant regions. However, obtaining

such dense annotations of bounding box and part annotations is labor-intensive, which limits both scalability and practicality of real-world fine-grained applications. Other methods [8,9,10,11] use an unsupervised learning scheme to localize informative regions. They eliminate the need for the expensive annotations, but how to focus on the right areas and use them is still worth investigating.

An overview of our MMAL-Net is shown in Fig. 1. Our method has three branches in training phase, whose raw branch mainly studies the overall characteristics of the object, and AOLM needs to obtain the object’s bounding box information with the help of the feature maps of the raw image from this branch. As the input of object branch, the finer scale of object image is very helpful for classification, because it contains the structural features of the target as well as the fine-grained features. Then, APPM proposes several part regions with the most disimilarity and less redundancy according to the feature maps of object image. The part branch sends the part image clipped from the object image to the network for training. It enables the network to learn fine-grained features of different parts in different scales. Unlike RA-CNN [12], the parameters of CNN and FC in our three branches are shared. Therefore, through the common learning process of the three branches, the trained model has a good classification ability for different scales and parts of object. In the testing phase, unlike RA-CNN [12] and NTS-Net [10] need to calculate the feature vectors of the multiple part region images and then concat these vectors for classification. After our repeated experiments, the best classification performance is simply obtained by the result of object branch, so our approach can reduce some calculations and inference time while achieving high accuracy.

Our main contributions can be summarized as follows:

- Our multi-branch network can be trained end-to-end and learn object’s discriminative regions for recognition effectively.
- The AOLM does not increase the number of parameters, so we do not need to train a proposal network like RA-CNN [12]. The accuracy of object localization is achieved by only using category labels.
- We present an attention part proposal method(APPM) without the need of part annotations. It can select multiple ordered discriminative part images, so that the model can effectively learn different scales parts’s fine-grained features.
- State-of-the-art performances are reported on three standard benchmark datasets, where our method stable outperforms the state-of-the-art methods and baselines.

2 Related works

In the past few years, the accuracy of benchmark on open datasets has been improved based on deep learning and fine-grained classification methods. They can be classified as follows: 1) By end-to-end feature encoding; 2) By localization-classification subnetworks.

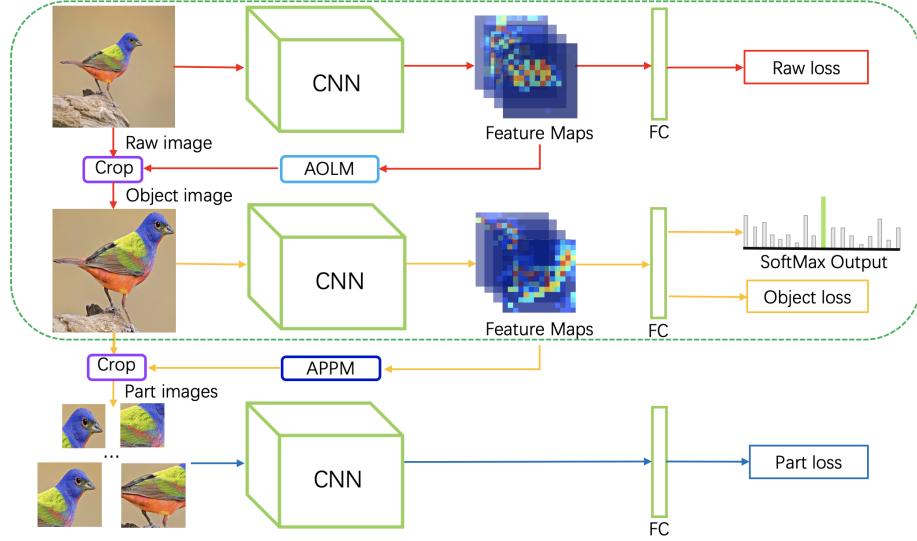


Fig. 1. The overview of our proposed MMAL-Net in the training phase. The red branch is raw branch, the orange branch is object branch, and the blue branch is part branch. In the dotted green box is the network structure for the test phase. The CNN (Convolutional Neural Networks) and FC (Fully Connection) layer of the same color represent parameter sharing. Our multi-branch all use cross entropy loss as the classification loss.

2.1 By End-to-End Feature Encoding

This kind of method directly learns a more discriminative feature representation by developing powerful deep models for fine-grained recognition. The most representative method among them is Bilinear-CNN [4], which represents an image as a pooled outer product of features derived from two bilinear models, and thus encodes higher order statistics of convolutional activations to enhance the mid-level learning capability. Thanks to its high model capacity, it achieves clear performance improvement on a wide range of visual tasks. However, the high dimensionality of bilinear features still limits its further generalization. In order to solve this problem, [13,14] try to aggregate low-dimensional embeddings by applying tensor sketching. They can reduce the dimensionality of bilinear features and achieve comparable or higher classification accuracy.

2.2 By Localization-Classification Subnetworks

This kind of method trains a localization subnetwork with supervised or weakly supervised to locate key part regions. Then the classification subnetwork uses the information of fine-grained regions captured by the localization subnetwork to further enhance its classification capability. Earlier works [5,6,7] belong to full supervision method depend on more than image-level annotations to locate

semantic key parts. [7] trained a region proposal network to generate proposals of informative image parts and concatenate multiple part-level features as a whole image representation toward final fine-grained recognition. However, maintaining such dense part annotations increases additional location labeling cost. Therefore, [8,9,10,11] take advantage of attention mechanisms to avoid this problem. There is no need of bounding box annotations and part annotations except image-level annotation.

3 Method

3.1 Attention Object Location Module (AOLM)

This method was inspired by SCDA [15]. SCDA uses a pre-trained model to extract image features for fine-grained image retrieval tasks. We improve it's positioning performance as much as possible through some measures. At the first, we describe the process of generating object location coordinates by processing the CNNs feature map as the Fig. 2 illustrated.

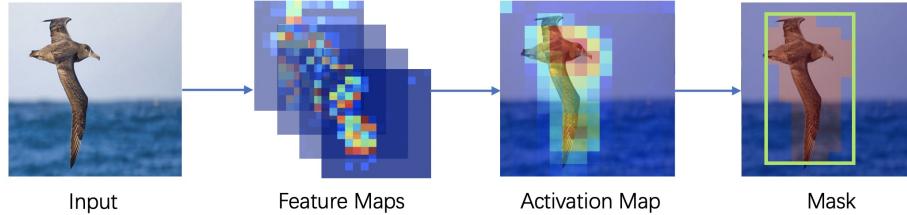


Fig. 2. The pipeline of the AOLM, we first get an activation map by aggregating the feature maps in the channel dimension, then obtain a bounding box according to activation map.

We use $F \in RC \times H \times W$ to represent feature maps with C channels and spatial size $H \times W$ output from the last convolutional layer of an input image X and f_i is the i-th feature map of the corresponding channel. As shown in Equ 1,

$$A = \sum_{i=0}^{C-1} f_i \quad (1)$$

activation map A can be obtained by aggregating the feature maps F . We can visualizes where the deep neural networks focus on for recognition simply and locates the object regions accurately from A. As shown in Equ 2, \bar{a} is the mean value of A .

$$\bar{a} = \frac{\sum_{x=0}^{W-1} \sum_{y=0}^{H-1} A(x, y)}{H \times W} \quad (2)$$

\bar{a} is used as the threshold to determine whether the element at that position in A is a part of object, and (x, y) is a particular position in a $H \times W$ activation map. Then we initially obtained a coarse mask map \widetilde{M}_{conv_5c} from the last convolutional layer $Conv_5c$ of ResNet-50 [16] according to Equ 3.

$$\widetilde{M}_{(x,y)} = \begin{cases} 1 & \text{if } A_{(x,y)} > \bar{a} \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

On the basis of the experimental results, we find that the object is often in the largest connected component of \widetilde{M}_{conv_5c} , so the smallest bounding box containing the largest connected area is used as the result of our object location. Only using a pre-trained VGG16 [17] in SCDA [15] achieved better position accuracy, but our pre-trained ResNet-50 does not reach a similar accuracy rate and dropped significantly. So we use the training set to train ResNet-50 for improving object location accuracy and experiments in subsection 4.5 verify the effectiveness of this approach. Then, inspired by [15] and [18] the performance of their methods all benefit from the ensemble of Multiple layers. So we get the activation map of the output of $Conv_5b$ according to Equ 1, which is one block in front of $Conv_5c$. Then we can get \widetilde{M}_{conv_5b} according to Equ 3, and finally we can get a more accurate mask M after taking the intersection of \widetilde{M}_{conv_5c} and \widetilde{M}_{conv_5b} . As shown in Equ 4.

$$M = \widetilde{M}_{conv_5c} \cap \widetilde{M}_{conv_5b} \quad (4)$$

Subsequent experimental results prove the effectiveness of these approaches to improve object location accuracy. This improved weakly supervised object location method can achieve higher localization accuracy than ACOL [19], ADL [20] and SCDA [15], without adding trainable parameters.

3.2 Attention Part Proposal Module(APPM)

Although AOLM can achieve higher localization accuracy, but there are some positioning results are part of the object. We improve the robustness of the model to this situation through APPM and part branch. It will be demonstrated in the next section. By observing the activation map A , we find that the area with high activation value of the activation map are often the area where the key part are located, such as the head area in the example. Using the idea of sliding window in object detection to find the windows with information as part images. Moreover, we implemented the traditional sliding window approach with full convolutional network to reduce the amount of calculation, just like Overfeat [21] gets the feature map of different windows from the feature map output from the previous branch. Then we aggregate each window's activation map A_w in the channel dimension and get its activation mean value \bar{a}_w according to Equ 5,

$$\bar{a}_w = \frac{\sum_{x=0}^{W_w-1} \sum_{y=0}^{H_w-1} A_w(x, y)}{H_w \times W_w} \quad (5)$$

H_w , W_w are the height and width of a window's feature map. We sort by the \bar{a}_w value of all windows. The larger the \bar{a}_w is, the larger the informativeness of this part region is, as shown in Fig. 3. However, we cannot directly select the first few windows, because they are often adjacent to the largest \bar{a}_w windows and contain approximate the same part, but we hope to select as many different parts as possible. In order to reduce region redundancy, we adopt non-maximum suppression (NMS) to select a fixed number of windows as part images with different scales. By visualizing the output of this module in Fig. 4, it can be seen that this method proposed some ordered, different importance degree part regions.

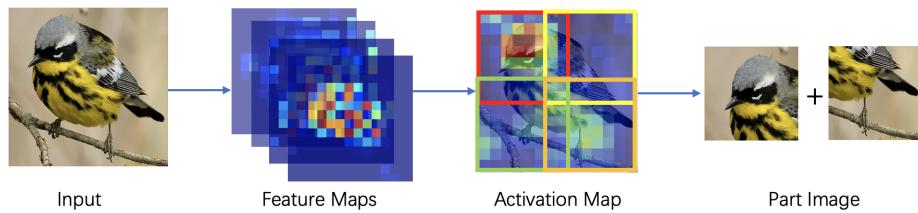


Fig. 3. The simple pipeline of the APPM. We use red, orange, yellow, green colors to indicate the order of windows' \bar{a}_w .

3.3 Architecture of MMAL-Net

In order to make the model fully and effectively learn the images obtained through AOLM and APPM. During the training phase, we construct a three branches network structure consisting of raw branch, object branch, and part branch, as shown in fig. 1. The three branches share a CNN for feature extraction and a FC layer for classification. Our three branches all use cross entropy loss as the classification loss. As shown in Equ 6, 7, and 8, respectively.

$$L_{raw} = -\log(P_r(c)) \quad (6)$$

$$L_{object} = -\log(P_o(c)) \quad (7)$$

$$L_{parts} = -\sum_{n=0}^{N-1} \log(P_{p(n)}(c)) \quad (8)$$

Where c is the ground truth label of the input image, P_r , P_o are the category probabilities of the last softmax layer output of the raw branch and object branch, respectively, $P_{p(n)}$ is the output of the softmax layer of the part branch corresponding to the n th part image, N is the number of part images. The total loss is defined as:

$$L_{total} = L_{raw} + L_{object} + L_{parts} \quad (9)$$

The total loss is the sum of the losses of the three branches, which work together to optimize the performance of the model during backpropagation. It enables the final convergent model to make classification predictions based on the overall structural characteristics of the object or the fine-grained characteristics of a part. The model has good object scale adaptability, which improves the robustness in the case of inaccurate AOLM localization. During the testing phase, we removed the part branch so as to reduce a large amount of calculations, so our method will not take too long to predict in practical applications. Due to our reasonable and efficient framework. MMAL-Net can achieves the state-of-the-art performance so far.

4 Experiments

4.1 Datasets

These three datasets are widely used as benchmarks for fine-grained classification(shown in Table 1). In our experiments, we only use the image classification labels provided by these datasets.

Table 1. Introduction of the three datasets used in this paper.

Datasets	Object	Class	Train	Test
CUB-200-2011(CUB) [1]	Bird	200	5994	5794
FGVC-Aircraft(AIR) [2]	Aircraft	100	6667	3333
Stanford Cars(CAR) [3]	Car	196	8144	8041

4.2 Implementation Details

In all our experiments, we first preprocess images to size 448×448 to get input image for raw branch and object branch. The object image is also scaled into 448×448 , but all part images are resized to 224×224 for part branch. We construct windows with three broad categories of scales: $\{[4 \times 4, 3 \times 5], [6 \times 6, 5 \times 7], [8 \times 8, 6 \times 10, 7 \times 9, 7 \times 10]\}$ for activation map of 14×14 size, and the number of a raw image's part images is $N = 7$, among them $N_1 = 2, N_2 = 3, N_3 = 2$. N_1, N_2 and N_3 are the number of three broad categories of scales windows mentioned above. ResNet-50 [16] pre-trained on ImageNet is used as the backbone of our network structure. During training and testing, we do not use any other annotations other than image-level labels. Our optimizer is SGD with the momentum of 0.9 and the weight decay of 0.0001, and a mini-batch size of 6 on a Tesla P100 GPU. The initial learning rate is 0.001 and multiplied by 0.1 after 60 epoch. We use Pytorch as our code-base.

4.3 Performance Comparison

We compared the baseline methods mentioned above on three commonly used fine-grained classification datasets. The experimental results are shown in the Table 2. By comparison, we can see that our method achieves the best accuracy currently available on these three datasets.

Table 2. Comparison results on three common datasets. Train Anno. represents using bounding box or part annotations in training.

Methods	Train Anno.	CUB	AIR	CAR
ResNet-50				
Bilinear-CNN [4]		84.1	84.1	91.3
SPDA-CNN [5]	✓	85.1	-	-
KP [14]		86.2	86.9	92.4
RA-CNN [12]		85.3	-	92.5
MA-CNN [9]		86.5	89.9	92.8
OSME+MAMC [22]		86.5	-	93.0
PC [23]		86.9	89.2	92.9
HBP [24]		87.1	90.3	93.7
Mask-CNN [6]	✓	87.3	-	-
DFL-CNN [25]		87.4	92.0	93.8
HSnet [7]	✓	87.5	-	-
NTS-Net [10]		87.5	91.4	93.9
MetaFGNet [26]		87.6	-	-
DCL [27]		87.8	92.2	94.5
TASN [11]		87.9	-	93.8
Ours		89.6	94.7	95.0

4.4 Ablation Studies

The ablation study is performed on the CUB dataset. Without adding any of our proposed methods, the ResNet-50 [16] obtained an accuracy of 84.5% under the condition that the input image resolution is 448×448 . In order to verify the rationality of the training structure of our three branches, we remove the object branch and part branch respectively. After removing the object branch, the best accuracy is 85.0% from the raw branch, a drop of 4.6%. This proves the great contribution of AOLM and object branch to improve the classification accuracy. After removing the part branch, the best accuracy is 87.3% from the object branch, down 2.3%. The results of the experiment show that part branch and APPM can improve the robustness of the model when facing AOLM unstable positioning results. The above experiment has shown that the three branches of our method all have a significant contribution to the final accuracy.

4.5 Object Localization Performance

Percentage of Correctly Localized Parts (PCP) metric is the percentage of predicted boxes that are correctly localized with more than 50% IOU with the ground-truth bounding boxes. On the CUB dataset, the best result of AOLM in terms of the PCP metric for object localization is 85.1%. AOLM clearly exceeds SCDA's [15] 76.8% and the recent weakly supervised object location methods ACOLs [19] 46.0% and ADLs [20] 62.3%. As shown in Table 3, the ensemble of multiple layers significantly improves object location accuracy. Through the experiment, we find that the object location accuracy using the pre-trained model directly is 65.0%. However, it rises to 85.1% after one iteration of training. And as the training progressed, CNN paid more and more attention to the most discerning region to improve classification accuracy which leads to localization accuracy drops to 71.1%. Even so, due to part branch and APPM make model has good adaptability to object's scale, it still achieved excellent classification performance.

Table 3. Object localization accuracy on CUB-200-2011.

Methods	localization Accuracy
ACOL [19]	46.0%
ADL [20]	62.3%
SCDA [15]	76.8%
AOLM(conv_5c)	82.2%
AOLM(conv_5b & conv_5c)	85.1%

4.6 Model and Inference Time Complexity

Unlike RA-CNN [12] and NTS-Net [10], the former has three branches with independent parameters and needs to train a subnetwork to propose finer scale images and the latter needs to train a navigator network to propose regions with large amount of information (such as different body parts of birds). Our MMAL-Net has some advantages in terms of parameter volume over them. First, the three branches parameters are shared, and secondly the AOLM and APPM modules do not require training, do not add additional parameter amounts. Thirdly their calculation amount is relatively smaller. Finally, better classification performance is achieved by MMAL-Net. Compared with the ResNet-50 baseline, our method yields a significantly better result (+4.1%) with the same parameter volume. As for inference time, RA-CNN needs to calculate the output of three branches and fuse them; NTS-Net needs to extract and fuse the 4 proposal local image features of the input image. Above reasons make their inference time relatively longer and our method has a shorter inference time. It only needs to calculate the output of the first two branches and does not need to fuse them, because the classification results are based on the output of the second branch

(object branch). For more accurate comparison, we conducted an inference time test on Tesla P100 and the input image size is 448×448 . The inference time of MMAL-Net and NTS-Net are summarized as follows: the running time on an image of NTS-Net is about 5.61 ms, and ours methods running time on an image is about 1.80 ms. Based on the above analysis, lower model and inference time complexity both add extra practical value to our proposed method.

4.7 Visualization of Object and Part Regions

In order to visually analyze the areas of concern for our AOLM and APPM, we draw the object's bounding boxes and part regions proposed by AOLM and APPM in Fig. 4. In the first column, we use red and green rectangles to denote the ground truth and predicted bounding box in raw image. It is very helpful for classification that the positioned areas of objects often cover an almost complete object. In columns two through four, we use red, orange, yellow, and green rectangles to represent the regions with the highest average activation values in different scales proposed by APPM, with red rectangle denoting the highest one. Fig. 4 conveys that the proposed area does contain more fine-grained information and the order is more reasonable on the same scale, which are very helpful for model's robustness to scale. We can find that the most discriminative regions of birds are head firstly , then is body, which is similar to human cognition.

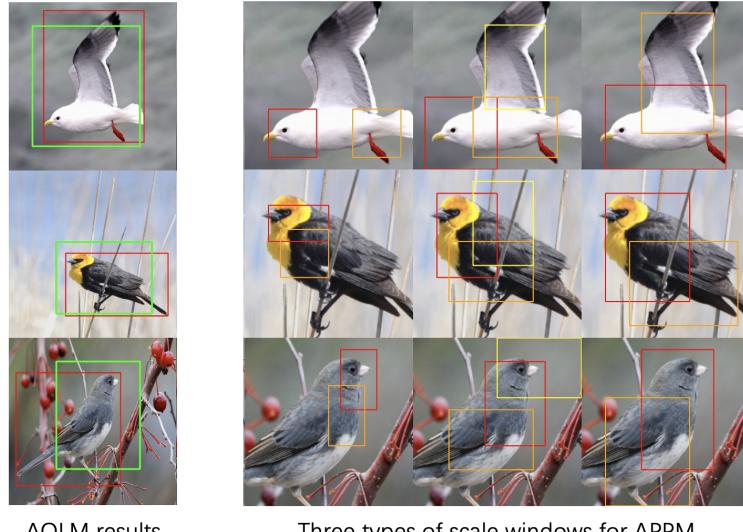


Fig. 4. Visualization of object and part regions.

5 Conclusion

In this paper, we propose an effective method for fine-grained classification without bounding box or part annotations. The multi-branch structure can make full use of the images obtained by AOLM and APPM to achieve excellent performance. Our algorithm is end-to-end trainable and achieves state-of-the-art results in CUB-200-2011 [1], FGVC Aircraft [2] and Stanford Cars [3] datasets. The future work is how to set the number and size of windows adaptively to further improve the classification accuracy.

References

1. 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)
2. Maji, S., Rahtu, E., Kannala, J., Blaschko, M., Vedaldi, A.: Fine-grained visual classification of aircraft. arXiv preprint arXiv:1306.5151 (2013)
3. Krause, J., Stark, M., Deng, J., Fei-Fei, L.: 3d object representations for fine-grained categorization. In: 4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13). Sydney, Australia (2013)
4. Lin, T.Y., RoyChowdhury, A., Maji, S.: Bilinear cnn models for fine-grained visual recognition. In: CVPR. pp. 1449–1457 (2015)
5. Zhang, H., Xu, T., Elhoseiny, M., Huang, X., Zhang, S., Elgammal, A., Metaxas, D.: Spda-cnn: Unifying semantic part detection and abstraction for fine-grained recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1143–1152 (2016)
6. 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)
7. Lam, M., Mahasseni, B., Todorovic, S.: Fine-grained recognition as hsnet search for informative image parts. In: CVPR. pp. 2520–2529 (2017)
8. Zhao, B., Wu, X., Feng, J., Peng, Q., Yan, S.: Diversified visual attention networks for fine-grained object classification. IEEE Transactions on Multimedia **19**(6), 1245–1256 (2017)
9. Zheng, H., Fu, J., Mei, T., Luo, J.: Learning multi-attention convolutional neural network for fine-grained image recognition. In: ICCV. pp. 5209–5217 (2017)
10. Yang, Z., Luo, T., Wang, D., Hu, Z., Gao, J., Wang, L.: Learning to navigate for fine-grained classification. In: ECCV. pp. 420–435 (2018)
11. Zheng, H., Fu, J., Zha, Z.J., Luo, J.: Looking for the devil in the details: Learning trilinear attention sampling network for fine-grained image recognition. In: CVPR. pp. 5012–5021 (2019)
12. Fu, J., Zheng, H., Mei, T.: Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition. In: CVPR. pp. 4438–4446 (2017)
13. Gao, Y., Beijbom, O., Zhang, N., Darrell, T.: Compact bilinear pooling. In: CVPR. pp. 317–326 (2016)
14. Cui, Y., Zhou, F., Wang, J., Liu, X., Lin, Y., Belongie, S.: Kernel pooling for convolutional neural networks. In: CVPR. pp. 2921–2930 (2017)

15. Wei, X.S., Luo, J.H., Wu, J., Zhou, Z.H.: Selective convolutional descriptor aggregation for fine-grained image retrieval. *IEEE T IMAGE PROCESS* **26**(6), 2868–2881 (2017)
16. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. pp. 770–778 (2016)
17. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
18. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: CVPR. pp. 3431–3440 (2015)
19. Zhang, X., Wei, Y., Feng, J., Yang, Y., Huang, T.S.: Adversarial complementary learning for weakly supervised object localization. In: CVPR. pp. 1325–1334 (2018)
20. Choe, J., Shim, H.: Attention-based dropout layer for weakly supervised object localization. In: CVPR. pp. 2219–2228 (2019)
21. Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., LeCun, Y.: Overfeat: Integrated recognition, localization and detection using convolutional networks. arXiv preprint arXiv:1312.6229 (2013)
22. Sun, M., Yuan, Y., Zhou, F., Ding, E.: Multi-attention multi-class constraint for fine-grained image recognition. In: ECCV. pp. 805–821 (2018)
23. Dubey, A., Gupta, O., Guo, P., Raskar, R., Farrell, R., Naik, N.: Pairwise confusion for fine-grained visual classification. In: ECCV. pp. 70–86 (2018)
24. Yu, C., Zhao, X., Zheng, Q., Zhang, P., You, X.: Hierarchical bilinear pooling for fine-grained visual recognition. In: Proceedings of the European conference on computer vision (ECCV). pp. 574–589 (2018)
25. Wang, Y., Morariu, V.I., Davis, L.S.: Learning a discriminative filter bank within a cnn for fine-grained recognition. In: CVPR. pp. 4148–4157 (2018)
26. Zhang, Y., Tang, H., Jia, K.: Fine-grained visual categorization using meta-learning optimization with sample selection of auxiliary data. In: Proceedings of the european conference on computer vision (ECCV). pp. 233–248 (2018)
27. Chen, S., Bai, Y., Zhang, W., Mei, T.: Destruction and construction learning for fine-grained image recognition. In: CVPR. pp. 5157–5166 (2019)