EXPLOITING COARSE-TO-FINE MECHANISM FOR FINE-GRAINED RECOGNITION

Yongzhong Wang[†‡], *Xu-Yao Zhang*[†], *Yanming Zhang*[†], *Xinwen Hou*[†], *Cheng-Lin Liu*[†]

[†]National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China
[‡]School of Automation, Hangzhou Dianzi University, Hangzhou, China

ABSTRACT

Fine-grained object recognition is more challenging than generic categorization due to the subtle difference between subcategories under the large intra-class pose change and appearance variations. The state-of-the-art fine-grained recognition methods usually utilize part detection or pose alignment to alleviate the pose variation, and then use convolutional neural networks (CNNs) to extract local discriminative features. Although the hierarchical structure of deep CNNs enables rich and discriminative visual feature extraction, the recognition methods so far mostly use the features of only the last convolutional layer for classification. In this paper, by exploiting the correlation of the convolutional features of within-layer and between-layer, we propose a method to integrate multi-layer convolutional features based on coarse-to-fine mechanism for improving the discrimination capability. Experiments on a number of public datasets show that the proposed method, without part annotation or pose alignment, yields superior or comparable performance to the state-of-the-art methods.

Index Terms— Fine-grained Recognition, CNNs, Coarse-to-fine, Cross-correlation, Autocorrelation

1. INTRODUCTION

Fine-grained recognition aims to distinguish different subcategory objects belonging to the same super-class, such as different types of birds [1,2], cars [3], plants [4,5], aircrafts [6], etc. These objects always have similar global shape or structure, and fine-grained recognition relies on identifying the subtle differences in appearance of certain local parts across related categories. However, these subtle inter-class differences are usually overwhelmed by large variations of pose, viewpoint, articulation, and clutter background.

In the past few years, the deep convolutional neural networks (CNNs) have shown outstanding performance in fine-grained recognition [7,8,9]. In order to overcome the large pose variation, most of these methods use pose alignment [10,11] or part detection [12,13,14] techniques in pre-processing, and then the last convolutional layer of CNNs model is generally employed to extract features from a local part or whole object [8,15]. As we know, though the last convolutional layer is most discriminative, after multiple convolution and pooling operations, the spatial resolution of feature maps in convolutional layers becomes smaller layer by layer. On the other hand, the receptive field of each unit in feature maps becomes larger layer by layer. For example, the VGGNet-E model [9] consists of 16 convolutional layers, five pooling layers and three fully-connected layers, its last convolutional layer just has a small spatial resolution of 7×7, and the receptive field of each unit of this layer is as large as 268×268. Considering that the input image size of VGGNet-E model is 224×224, the responses of the last convolutional layer are very global. Only using the last convolutional layer features cannot distinguish local details between different subcategories sufficiently. Contrarily, the responses of previous convolutional layers are more local and include more details, but lack discrimination in object level than last convolutional layer.

In this work, we investigate the complementarities between hierarchical convolutional layers, and propose a novel method that separately employ within-layer autocorrelation and between-layer cross-correlation to integrate multiple hierarchical convolutional layers via a coarse-to-fine strategy. Our experimental results show that: compared to classification with only last convolutional layer features, the proposed method can extract more local discriminative features and significantly improve fine-grained recognition performance.

2. RELATED WORK

In recent years, many different methods have been proposed to deal with a variety of fine-grained recognition tasks [10,11,14]. For extracting the subtle and discriminative features between subcategories, the state-of-the-art fine-grained recognition system commonly follow the pipeline of part detection or pose normalization, and discriminative feature extraction by convolutional neural networks.

In fine-grained case, a large pose variation can cause drastic changes of object appearance, compared to the slight differences between subcategories. Therefore, many methods

This work was supported by the National Basic Research Program of China Grant 2012CB316302, the National Natural Science Foundation of China Grants 61273289, 61403380, 61375039, and Zhejiang Provincial Nature Science Foundation grant No. LY15F030019.

utilize pose alignment or part detection to overcome the pose or viewpoint changes before extracting object features. For example, Gavves et al. [11] propose to roughly align objects based on fitting the object segmentation results as an ellipse, the alignments are used to find corresponding parts between different object images and transfer part annotation from training images to test images. Recently, Zhang et al. [14] employ the R-CNN framework [16], which is trained with annotated object parts, to detect parts and the whole object, for obtaining a pose-normalized object representation. Bronson et al. [10] align the detected keypoints to the corresponding keypoints of a prototype to realize pose normalization, and then apply CNNs model to extract part features. Goering et al. [17] search for training images that have a similar shape to the current test image, and then directly transfer the part annotations of training image to test image. Krause et al. [18] define a graph model to find the corresponding regions of different objects based on co-segmentation results of all images without using part annotation. In general, due to the limitation of training samples in fine-grained recognition, various pose changes and flexible transformations, the part detection or pose normalization may not be accurate enough for determining the locations of subtle differences between subcategories.

In the past few years, deep convolutional neural networks have achieved great success in fine-grained recognition [18,19], just like in many other computer vision tasks [8,20]. Most of these methods in fine-grained domain typically use the output of last convolutional layer as a feature representation, which takes advantage of the powerful capability of CNNs in extracting features from the pose normalized images [11,21] or detected parts [14,22]. Lin et al. [19] trained a bilinear CNNs model consisting of two convolutional neural networks for fine-grained recognition without using part annotations, and last convolutional layers of these two CNNs models is used to extract object features. Liu et al. [23] use the feature maps of succeeding convolutional layer as indicator maps to pool the subarrays of feature maps of previous convolutional layer as local features, and then concatenate all the local features as final image representation. Xiao et al. [24] propose to apply three types of visual attention to fine-grained recognition, and the attention is derived from the CNNs model trained with classification task. Recently, Hariharan et al. [15] stack the outputs of all units above a given location at all layers of CNNs into one vector named hypercolumns, which significantly improve the state-of-the-art results on three fine-grained localization tasks, and demonstrates that previous layers of CNNs are useful for more precise localization.

Compared with the above models, our method employs only one convolutional neural networks and integrates the different convolutional layers based on a coarse-to-fine mechanism, to simultaneously extract global and local discriminative features for fine-grained recognition. To the best of our knowledge, this is the first work that employs a coarse-to-fine mechanism to combine the complementarities between convolutional layers to improve fine-grained recognition.

3. PROPOSED METHOD

3.1. Coarse-to-Fine Mechanism

Commonly, the last layer of a CNNs model includes more discriminative semantic information and is more robust to various disturbing factors such as pose, articulation, illumination, location and so on. However, after a series of convolution and pooling operations, the last convolutional layer has low spatial resolution, and the receptive field of each unit in last convolutional layer becomes very large. It implies that the feature representation of last convolutional layer is too global or coarse to capture the local detailed differences between subcategories for fine-grained recognition. In other words, the features of last convolutional layer may be a good enough representation for generic object classification but not for fine-grained recognition. It is well known that the human vision system has a coarse-to-fine mechanism during visual information processing [25], coarse spatial responses convey global information about an image, while fine spatial response carry more detailed information. The coarse-to-fine scheme enables the use of coarse scale information to disambiguate matching information in finer scales. Obviously, for fine-grained recognition, not only the global semantic information but also the local details are required to achieve high accuracy.

Benefiting from the hierarchical structure and shared weights, a deep convolution neural networks can extract rich representations of object image with increasing level of abstraction, from low-level image structure to high-level semantic features. In this work, we aim to keep the semantic discriminative power of last convolutional layer, and meanwhile exploit previous convolution layers to extract more detailed features. Specifically, considering the cross-correlation of two vectors can calculate all combinations of two components belonging to these two vectors separately, we adopt the cross-correlation between the feature vectors of last convolution layer (coarse layer) and previous ones (fine layer), which have the same receptive field, to extract global discrimination information and local details. That means if a unit of the coarse layer has a strong response, in the cross-correlation feature its corresponding local features of the fine layer vector will be enhanced. The scheme of coarse-to-fine

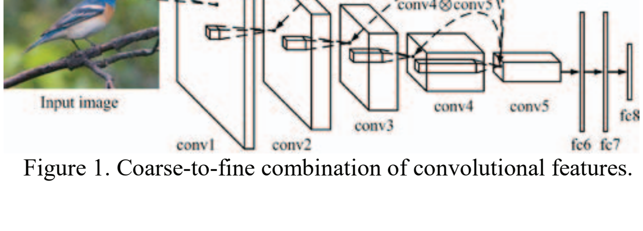

Figure 1. Coarse-to-fine combination of convolutional features.

combination of multilayer convolutional features is shown in Fig. 1, where "conv4 ⊗ conv5" means the cross-correlation between the 4th and 5th convolutional layers.

3.2. Cross-correlation of Convolutional Layers

In CNNs model, each channel of a feature map can be deemed as the response of a special part detector. Let $\mathbf{u}_{ij} \in R^{n_i}$ denotes the convolutional feature vector of the j-th unit in the i-th convolution layer consisted of n_i channels. When the q-th element value of $\mathbf{u}_{ij}$ is large, it implies that the feature or pattern detected by the q-th part detector of layer i is more salient in the receptive field of unit j. If we want to furthermore explore the finer details of the q-th pattern, the responses of earlier convolutional layers corresponding to the same receptive field of the unit j of layer i should be used. As mentioned before, the cross-correlation is used to convey the information from coarse scale to disambiguate matching information on finer scales, which calculates by the outer product between two different convolutional feature vectors as follows,

$$F_{ik}^{c} = \frac{1}{m_i}\sum_{j=1}^{m_i}\mathbf{u}_{ij} \otimes \mathbf{u}_{kj}^{p}, \tag{1}$$

where ⊗ denotes outer product, F_{ik}^{c} is cross-correlation coefficient matrix between the convolutional layer i and k, and m_i is the number of units of the convolutional layer i. $\mathbf{u}_{kj}^{p}$ is the convolutional feature vector corresponding to $\mathbf{u}_{ij}$ in the previous convolutional layer k. It is computed by sum pooling over the local region $r_k(j)$ in the convolutional layer k, which has the same receptive field as $\mathbf{u}_{ij}$.

$$\mathbf{u}_{kj}^{p} = \frac{1}{|r_k(j)|}\sum_{l \in r_k(j)}\mathbf{u}_{kl}, \tag{2}$$

where $|r_k(j)|$ denotes the number of units in this local region.

3.3. Autocorrelation of Convolutional Layer

In the same convolutional layer, the response values of a channel describe how likely the object image possesses a certain pattern in every unit, and how salient the pattern is. Obviously, the statistics of co-occurrence of different patterns in different channels will be helpful to distinguish the subtle differences between subcategories. Therefore, in addition to the cross-correlation features, we adopt autocorrelation features within the same convolutional layer to further improve the discriminability of convolutional feature. The autocorrelation matrix F_i^a is calculated by

$$F_i^{a} = \frac{1}{m_i}\sum_{j=1}^{m_i}\mathbf{u}_{ij} \otimes \mathbf{u}_{ij}. \tag{3}$$

Because the autocorrelation matrix is symmetric, we use the upper or lower triangular part of the matrix as autocorrelation feature.

3.4. Combinations of Features

In this section, we validate the effectiveness of the proposed cross-correlation and autocorrelation features, and identify their best combination through experiments. In validation experiments, we use the VGGNet-E model [9] pre-trained on ImageNet [26] to extract convolutional features of object image within a given bounding box, and perform $l2$ normalization over each feature map separately to alleviate the impact of large difference of response magnitude between different convolutional layers. Before calculating the outer product, the dimensionality of convolutional feature of each unit is reduced to 100 using the principal component analysis (PCA) technique, and then the cross-correlation features and autocorrelation features are power normalized [27] with power coefficient of 0.5. After extracting object features, the linear SVM [28] is used as final classifier.

Tab. 1 gives the results of a variety of autocorrelation features, cross-correlation features and their combinations on CUB-2011 data set [2]. "C_5_S" denotes the feature of the 5th (last) convolutional layer with spatial pyramid matching (SPM) kernel [29]. "C_3+C_4+C_5" is obtained by directly concatenating the convolutional feature vectors of the 3th, 4th and 5th layers after sum pooling separately, like used in [15]. The results show that the straight combination of different convolutional layer features, like "C_3+C_4+C_5", cannot improve the recognition performance. Compared with this, the autocorrelation features is remarkably superior to convolutional feature of the same convolutional layer, such as $\boldsymbol{F}_5^a$ and C_5_S, although the dimensionality of C_5_S is also very high. The cross-correlation features benefiting from the coarse-to-fine strategy is better than the autocorrelation features, such as $\boldsymbol{F}_{54}^c$ than $\boldsymbol{F}_4^a + \boldsymbol{F}_5^a$, $\boldsymbol{F}_{53}^c + \boldsymbol{F}_{54}^c$ than $\boldsymbol{F}_3^a$+$\boldsymbol{F}_4^a$+$\boldsymbol{F}_5^a$. In addition, it should be noted that there is strong complementarity between the autocorrelation features and the cross-correlation features. For example, the recognition accuracy of using $\boldsymbol{F}_4^a + \boldsymbol{F}_5^a$ is significantly better than only using $\boldsymbol{F}_5^a$, $\boldsymbol{F}_{53}^c + \boldsymbol{F}_{54}^c$ is also better than $\boldsymbol{F}_{54}^c$, and $\boldsymbol{F}_4^a$+$\boldsymbol{F}_5^a$+$\boldsymbol{F}_{53}^c$+$\boldsymbol{F}_{54}^c$ is obviously superior to $\boldsymbol{F}_{53}^c$+$\boldsymbol{F}_{54}^c$ and $\boldsymbol{F}_3^a$+$\boldsymbol{F}_4^a$+$\boldsymbol{F}_5^a$. The CL45 method [23] use the same 4th and 5th convolutional layer features of pre-trained CNNs model on ImageNet dataset as our $\boldsymbol{F}_{54}^c$ feature, but the recognition accuracy of using $\boldsymbol{F}_{54}^c$ is obviously better that the CL45C method, which demonstrates that our method using coarse-to-fine mechanism is more effective. In all combinations, the $\boldsymbol{F}_4^a$ + $\boldsymbol{F}_5^a$ + $\boldsymbol{F}_{53}^c$ + $\boldsymbol{F}_{54}^c$ feature is used in the sequential experiments, considering this combination has better performance and modest dimensionality.

Table1. Comparison of different combination features

Features	C_5_S	C_3+C_4+C_5	CL45 [23]	F_5^a
mAP %	71.9	71.5	73.5	74.2
F_4^a+F_5^a	F_{54}^c	F_3^a+F_4^a+F_5^a	F_{53}^c+F_{54}^c	F_4^a+F_5^a+F_{53}^c+ F_{54}^c
77.9	78.2	78.4	79.5	**80.3**

4. EXPERIMENTS

Figure 2. Examples for CUB-2011 dataset (left), cars-196 dataset (center), aircraft-100 dataset (right).

We compare the performance of the proposed method with state-of-the-art approaches on the CUB-2011 [2], cars-196 [3] and aircrafts-100 [6] datasets. Fig. 2 shows some example images from these datasets. We fine-tune the pre-trained VGGNet-E model for each classification task. The training data is not augmented except for random flips. In these experiments, the dimensionalities of convolutional layer features of the 5th, 4th and 3th convolutional layers are reduced by PCA to 400, 200 and 100, respectively. We use mean average precision (mAP) as the accuracy measure of classification, and at test time, all methods just use the test images, without using their flipped copies [19], for a fair comparison.

In Tab. 2, we compare our method to several state-of-the-art methods on CUB-2011, listing the annotation each method used and whether the pose normalization technique is adopted. Our method achieves the best performance by a large margin among the methods which use no part annotation, and even outperforms some methods that use part annotations, only is beaten by the variant of PN-DCN [10], which uses not only part annotation but also complicated pose alignment technique. It should be noted that PD+DCoP [18] and B-CNN [19] employ the same CNNs architecture as ours, and also use the bounding box annotations during training and testing. B-CNN uses two separate CNN models (VGGNet-E and VGGNet-M) to model "where" the parts are and "what" the parts look like, and requires the input image size is double than that we used. Our method uses only one CNNs model and doesn't utilize any pose alignment and part detection approaches.

In Tab. 3 we compare the performance of the proposed method with several related methods on cars-196 dataset.

Table 2. Comparison of methods on CUB-2011

Method	Oracle Parts	Oracle BBox	Pose Alignment	mAP %
DPD+DeCAF [7]	√	√		65.0
POOF [12]	√	√		73.3
PB R-CNN [14]	√	√		82.0
PN-DCN [10]	√	√	√	**85.4**
PD+DCoP [18]		√	√	82.8
B-CNN [19]		√		81.9
Ours		√		**83.7**

Table 3. Comparison of methods on cars-196

Method	Oracle BBox	Pose Alignment	Part Detection	mAP %
R-CNN [18]	√		√	88.4
Symbiotic [22]	√			78.0
Adap_FV [30]	√			82.7
B-CNN [19]				88.2
PD+DCoP[18]	√	√		**92.8**
Ours	√			**91.5**

Because the cars-196 dataset has no part annotation, many methods that require part annotations during training or testing cannot be used on this dataset. The recognition accuracy of our proposed method is close to PD+DCoP [18], and significantly outperforms the other method. Among these methods, R-CNN [18], B-CNN [19], and PD+DCoP all use the fine-tuned VGGNet-E model like ours. The precision of our method is slightly lower than PD+DCoP. One reason may be that on all datasets we use the same parameters to fine-tune the CNNs model, if with more careful fine-tuning our method would have better results.

In Tab. 4, the classification results of our method against the state-of-the-art classification algorithms are compared on aircraft-100 dataset. ELLF-CNN [31] first uses the segmented and aligned images to find important parts for classification, and then employs a modified Krizhevsky's CNNs model [32] to learn appearance descriptors. The symbiotic method [22] trains a symbiotic part localization and segmentation model for part-localization, and encodes the foreground and the box of each part by color histogram and fisher vector. Our method outperforms all the previous methods by a large margin.

Table 4. Comparison of methods on aircraft-100

Method	Oracle BBox	Pose Alignment	Part Detection	mAP %
ELLF-CNN [31]	√		√	73.9
Symbiotic [22]	√		√	72.5
Adap_FV [30]	√			80.7
B-CNN [19]				79.4
Ours	√			**87.6**

5. CONCLUSION

In this work, we combine the hierarchical convolutional layers of CNNs model based on a coarse-to-fine mechanism to get integrated features with global discrimination and local details. Our experiments show that this combination yields superior or comparable results to the state-of-the-art methods without using any part detection or pose normalization techniques. In the future work, we will explore some training strategies to strengthen the discrimination of intermediate convolutional layers apart from last convolutional layer of CNNs according to the cross-correlation features.

REFERENCES

[1] T. Berg, J. Liu, S. W. Lee, M. L. Alexander, D. W. Jacobs, and P. N. Belhumeur, "Birdsnap: large-scale fine-grained visual categorization of birds," In *CVPR*, IEEE, 2014.

[2] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, P. Perona, "Caltech-UCSD birds 200," *California Institute of Technology*. CNS-TR-2010-001, 2010.

[3] J. Krause, M. Stark, J. Deng, F. Li, "3D object Representations for fine-grained categorization," In *IEEE Workshop on 3D Representation and Recognition*, IEEE, 2013.

[4] A. Angelova, S. Zhu, and Y. Lin, "Image segmentation for large-scale subcategory flower recognition," In *Workshop on Applications of Computer Vision*. IEEE, 2013.

[5] N. Kumar, P. N. Belhumeur, Biswas, et al., "Leafsnap: a computer vision system for automatic plant species identification," In *ECCV*, Springer, 2012.

[6] S. Maji, J. Kannala, E. Rahtu, M. Blaschko, and A. Vedaldi, "Fine-grained visual classification of aircraft," *Technical report*, 2013.

[7] 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," *arXiv preprint arXiv:1310.1531*, 2013.

[8] A. S. Razavin, H. Azizpour, J. Sullivan, and S. Carlsson, "CNN features off-the-shelf: An astounding baseline for recognition," In *DeepVision workshop*, 2014.

[9] K. Simonyan, and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[10] S. Branson, G. Van Horn, P. Perona, and S. Belongie, "Improved bird species recognition using pose normalized deep convolutional nets," In *BMVC*, 2014.

[11] E. Gavves, B. Fernando, C. G. Snoek, A. W. Smeulders, and T. Tuytelaars, "Fine-grained categorization by alignments," In *ICCV*, IEEE, 2013.

[12] T. Berg, P. N. Belhumeur, "Poof: Part-based one-vs.-one features for fine-grained categorization, face verification, and attribute estimation," In *CVPR*, IEEE, 2013.

[13] N. Zhang, R. Farrell, F. Iandola, and T. Darrell, "Deformable part descriptors for fine-grained recognition and attribute prediction," In *ICCV*, IEEE, 2013.

[14] N. Zhang, J. Donahue, R. Girshick, and T. Darrell, "Part based R-CNNs for fine-grained category detection," In *ECCV*. Springer, 2014.

[15] B. Hariharan, P. Arbelaez, R. Girshick, and J. Malik, "Hypercolumns for Object Segmentation and Fine-grained Localization," In *CVPR,* IEEE, 2015.

[16] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," In *CVPR*, IEEE, 2014.

[17] C. Goering, E. Rodner, A. Freytag, and J. Denzler, "Nonparametric part transfer for fine-grained recognition," In *CVPR*, IEEE, 2014.

[18] J. Krause, H. Jin, J. Yang, F. Li, "Fine-grained recognition without part annotations," In *CVPR*, IEEE, 2015.

[19] T. Lin, A. R. Chowdhury, S. Maji, "Bilinear CNN models for fine-grained visual recognition," *arXiv preprint arXiv: 1504.07889 v1*, 2015.

[20] K. He, X. Zhang, S. Ren, J. Sun, "Deep residual learning for image recognition," *arXiv preprint arXiv:1512.03385*, 2015.

[21] R. Farrell, O. Oza, N. Zhang, V. I. Morariu, T. Darrell, and L. S. Davis, "Birdlets: Subordinate categorization using volumetric primitives and pose-normalized appearance," In *ICCV*, IEEE, 2011.

[22] Y. Chai, V. Lempitsky, and A. Zisserman, "Symbiotic segmentation and part localization for fine-grained categorization," In *ICCV*, IEEE, 2013.

[23] L. Liu, C. Shen, A. van den Hengel, "The treasure beneath convolutional layers: cross convolutional layer pooling for image classification," In *CVPR*, IEEE, 2015.

[24] T. Xiao, Y. Xu, K. Yang, J. Zhang, Y. Peng, and Z. Zhang, "The application of two-level attention models in deep convolutional neural network for fine-grained image classification," In *CVPR*, IEEE, 2015.

[25] M. Bar, "Visual objects in context," *Nature Reviews Neuroscience*, 5(8), pp. 617-629, 2004.

[26] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al., "Imagenet large scale visual recognition challenge," a*rXiv preprint arXiv:1409.0575*, 2014.

[27] F. Perronnin, J. Sanchez, T. Mensink, "Improving the fisher kernel for large-scale image classification," In *ECCV*, Springer, 2010.

[28] R. E. Fan, K. W. Chang, C. J. Hsieh, X. R. Wang, C. J. Lin, "LIBLINEAR: A library for large linear classification," *Journal of Machine Learning Research*, 9(2008), pp.1871-1874, 2008.

[29] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," In *CVPR*, IEEE, 2006.

[30] P. H. Gosselin, N. Murray, H. Jegou, and F. Perronnin, "Revisiting the fisher vector for fine-grained classification," *Pattern Recognition Letters*, 49, pp.92-98, 2014.

[31] J. Krause, T. Gebru, J. Deng, L. Li, F. Li, "Learning Features and Parts for Fine-Grained Recognition," In *ICPR*, 2014.

[32] A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," In *NIPS*, 2012.