FINE-GRAINED VISUAL CATEGORIZATION WITH FINE-TUNED SEGMENTATION

Lingyun Li[1], Yanqing Guo[1], Lingxi Xie[2], Xiangwei Kong[1], Qi Tian[3]

[1]Dalian University of Technology, Dalian, Liaoning 116024, China
[2]Dept. of Computer Science and Technology, Tsinghua University, Beijing 100084, China
[3]Dept. of Computer Science, University of Texas at San Antonio, TX 78249, USA

ABSTRACT

Fine-grained visual categorization (FGVC) refers to the task of classifying objects that belong to the same basic-level class (*e.g.*, different *bird* species). Since the subtle inter-class variation often exists on small parts (*e.g.*, *beak*, *belly*, *etc.*), it is reasonable to localize semantic parts of an object before describing it. However, unsupervised part-segmentation methods often suffer from over-segmentation which harms the quality of image representation. In this paper, we present a fine-tuning approach to tackle this problem. To this end, we perform a greedy algorithm to optimize an intuitive objective function, preserving principal parts meanwhile filtering noises, and further construct mid-level parts beyond the refined parts toward a more descriptive representation. Experiments demonstrate that our approach achieves competitive classification accuracy on the CUB-200-2011 dataset with both Fisher vectors and deep conv-net features.

Index Terms— Fine-Grained Visual Categorization, Part-based Model, Object Segmentation, Refinement.

1. INTRODUCTION

Fine-grained visual categorization (FGVC) refers to the task of distinguishing subordinate categories (*e.g.*, *tree sparrow*, *Ivory gull*, *Anna hummingbird*, *etc.*) which belong to the same basic-level category (*bird*). The subtle inter-class variation is often the major challenge of FGVC.

The Bag-of-Features (BoF) model is widely adopted for image classification. It extracts local descriptors, encodes and summarizes them into a global image representation. Sometimes, spatial context modeling is adopted to group descriptors according to their coordinates on the image. To introduce more visual clues based on parts, unsupervised part detectors are proposed for fine-grained tasks. Template matching models are adopted to automatically discover object parts [1] [2], and the Deformable Part Model (DPM) is verified efficient for part alignment [3] [4]. Researchers also suggest to partition the segmented foreground into parts in both supervised [5] and unsupervised [6] manners. However, unsupervised part detectors [4] [6] often suffer from over-segmentation, which leads to ambiguous image representation and, consequently, unsatisfied classification accuracy. An over-segmented example is shown in the upper-right part of Figure 1.

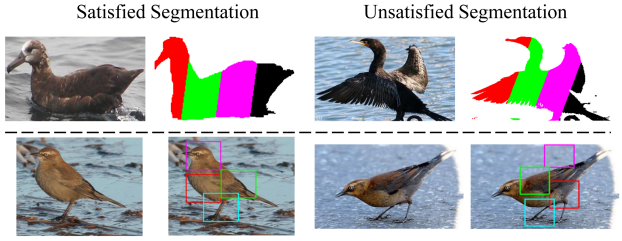

Fig. 1: Sample images from the CUB-200-2011 dataset [7] (best viewed in color). Each image is cropped with the provided bounding box. **Top**: Examples of fine-grained alignment [6]. **Bottom**: Examples of symbiotic segmentation and part localization [4].

In this paper, we propose a simple fine-tuning algorithm to combat over-segmentation. Based on a straightforward intuition, we formulate the fine-tuning process with an objective function, and optimize it using a greedy algorithm. We further construct mid-level visual concepts on the basis of the refined parts with a bruteforce search. It is verified that, although the number of parts is decreased during mergence and combination, higher classification accuracy is achieved, implying that more discriminative image representation is obtained. The main contribution of this paper is to provide an evidence on the benefit of fine-tuned segmentation for fine-grained visual categorization. We evaluate our algorithm with a *bird* classification task on the CUB-200-2011 dataset [7], and demonstrate competitive performance, *i.e.*, 65.13% with Fisher vectors and 70.34% with deep conv-net features.

2. RELATED WORKS

Fine-grained visual categorization (FGVC) is aimed at discriminating images of the same basic-level concept, such as *flower* [8], *aircraft* [9], *dog* [10] and *bird* [7]. It is closely related to two well studied topics in computer vision, *i.e.*, image representation and object part detection.

2.1. Image Representation Models

There are lots of properties shared by fine-grained and generic image classification, in which images are represented as high-dimensional vectors and fed into generalized machine learning algorithms for training and testing. Popular methods for image representation include Bag-of-Features (BoF) model and deep Convolutional Neural Networks (CNNs).

The BoF model starts from extracting local descriptors. SIFT [11], HOG [12] and Local Color Statistics (LCS) [13] are widely adopted, and dense sampling is efficient for image classification. Next, a codebook is trained to estimate feature distribution, and descriptors are encoded as compact feature vectors. Popular feature encoding methods include LLC [14] and Fisher vectors [15]. Pooling algorithms [16] [17] are used to aggregate local feature vectors into a global representation.

In recent years, CNNs [18] have been incorporated into a number of visual categorization systems. With the availability of large-scale image data [19] and powerful computational resources, CNN has been verified to produce superior image classification performance to conventional BoF models. The intermediate responses of a pre-trained CNN model could also be applied to other image applications [20], *e.g.*, in classification tasks, deep conv-net features significant outperform conventional handcrafted descriptors [21] [22] [23].

2.2. Part Detection

Fine-grained visual categorization often involves discriminating objects by subtle inter-class differences. To this end, semantic part detection is generally adopted.

One solution is foreground segmentation [8] [24] [25] followed by part detection [1] [2] [6]. It is shown that filtering out background significantly improves classification accuracy [8]. Meanwhile, various techniques are adopted to improve the detection quality, such as co-segmentation [24] and parts derived from foreground shapes [6] [25].

There are also efforts aimed at training unsupervised algorithms for part detection [26] [27] [28], or performing segmentation and part localization in a joint manner [4]. Either model provides better alignment for image description. In [4], Chai *et al.* apply a symbiotic set of templates to perform foreground segmentation and part detection simultaneously.

3. THE PROPOSED ALGORITHM

3.1. Motivation

We propose an approach to fine-tune unsupervised part segmentation. We start from the segmentation results of [6], which produces state-of-the-art classification performance. After foreground detection, an ellipse is fitted to the object and the main direction is defined by the major axis of the ellipse. The object is partitioned uniformly along the main direction into four **principal** parts, as shown in Fig. 2 (a).

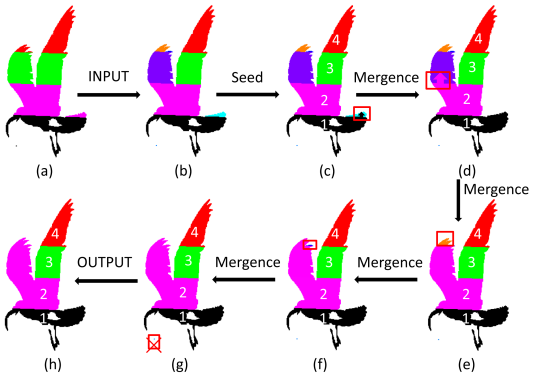

Fig. 2: An illustration of fine-tuned segmentation. (a) Original segmentation. (b) Independent regions. (c) Four seed parts which are the largest regions with different indicator value. (d)-(f) Iteratively merging small regions into connected seed parts. (g) Discarding isolated regions. (h) Refined parts.

We point out that such rough segmentation often fails to satisfy some straightforward properties, *e.g.*, intuitively, all the segmented parts should be **connected**. Empirically, fractional regions in segmentation often correspond to less meaningful visual contents which should be merged into larger regions. In the following, we design a fine-tuning algorithm to alleviate the ambiguity caused by over-segmentation.

3.2. Merging Small Fractions

We define an indicator $a_{ij} \in \{0, 1, 2, 3, 4\}$ for the pixel at position (i, j). Pixels with $a_{ij} = 0$ belong to the background. We call a region **independent** if it is connected and contains pixels with the same nonzero indicator value. We denote the number of **maximal independent regions** as L, most often $L > 4$, and denote each region as $\mathcal{R}_l$ which is the set of the contained pixels, for $l = 1, 2, \ldots, L$. The union of original regions is denoted as $\mathcal{R}$: $\mathcal{R} = \bigcup_{l=1}^{L} \mathcal{R}_l$. Let $s_l = |\mathcal{R}_l|$ and $v_l \in \{1, 2, 3, 4\}$ denote the size and the indicator value of the l-th region, respectively. The goal of refinement is to obtain four connected refined parts meanwhile maximally preserving original segmentation results, *i.e.*, the indicator values of a maximal number of pixels are unchanged. We denote v'_l for the indicator value of the l-th region after refinement, then the loss function of refinement could be formulated as:

$$L(\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3, \mathcal{P}_4) = \sum_{i=1}^{4} \sum_{\mathcal{R}_l \in \mathcal{P}_i} s_l \delta_l, \tag{1}$$

$$\text{s. t. } \mathcal{P}_i \text{ is connected, } i = 1, 2, 3, 4, \tag{2}$$

where

$$\mathcal{P}_i = \bigcup_{v'_l = i} \mathcal{R}_l, i = 1, 2, 3, 4, \text{ and } \delta_l = \begin{cases} 0, & v'_l = v_l \\ 1, & v'_l \neq v_l \end{cases}. \tag{3}$$

Fine-tuned parts are obtained by optimizing (1):

$$\{\mathcal{P}_1^\star, \mathcal{P}_2^\star, \mathcal{P}_3^\star, \mathcal{P}_4^\star\} = \arg\min_{\{\mathcal{P}_1,\mathcal{P}_2,\mathcal{P}_3,\mathcal{P}_4\}} L. \quad (4)$$

One can easily observe that the original regions would not be divided in refinement, *i.e.*, if two pixels belong to the same region $\mathcal{R}_l$, they will definitely fall into the same part $\mathcal{P}_i$. Therefore, it is reasonable to assume that the maximal region with the indicator value i are not changed ($i = 1, 2, 3, 4$). These four largest regions are selected as **seed parts**, and other regions are merged with them to preserve the connection constraint (2). The motivation of mergence lies in that neighboring regions often share very similar visual contents.

Four seed parts are removed from the union firstly: $\mathcal{R} \leftarrow \mathcal{R} - \bigcup_{i=1}^4 \mathcal{P}_i$. Then, we attempt to absorb smaller regions which are connected to seed parts. A simple greedy algorithm is adopted here. If a smaller region $\mathcal{R}_l \subset \mathcal{R}$ is connected to at least one seed part, it would be absorbed by the largest connected seed part $\mathcal{P}_i$, *i.e.*, $\mathcal{P}_i \leftarrow \mathcal{P}_i \bigcup \mathcal{R}_l$, $v_l' \leftarrow v_i$. And the absorbed region is excluded from the region union: $\mathcal{R} \leftarrow \mathcal{R} - \mathcal{R}_l$. This process will be iterated until $\mathcal{R} = \varnothing$ or all the remaining regions in $\mathcal{R}$ are only connected to background. These regions would be discarded (set as background). Finally, we obtain four refined parts $\{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3, \mathcal{P}_4\}$. Combining these four parts yields the refined foreground. An example of mergence is illustrated in Figure 2.

3.3. Constructing Mid-level Parts

Following the idea of [5], we construct mid-level parts which might contain more descriptive visual contents. Starting from the refined parts $\{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3, \mathcal{P}_4\}$, mid-level parts are constructed by a bruteforce search. Inspired by the intuition that neighboring parts often share similar properties, we combine neighboring parts into a larger part and justify the performance. Let $\mathcal{C}_{mn} = \{\mathcal{P}_m \bigcup \mathcal{P}_n\}$ denote a mid-level part, where m, n are the indices of two neighboring parts. Since the number of parts is relatively small ($P = 4$) in this case, the bruteforce search could finish in a very short time. We observe that the best classification accuracy is obtained with a combined part $\mathcal{C}_{23}$ meanwhile discarding both $\mathcal{P}_2$ and $\mathcal{P}_3$ ($\mathcal{P}_1$ and $\mathcal{P}_4$ are remain unchanged). Please refer to Sec. 4.2 for details. This result reveals that mid-level parts might be more discriminative since along-boundary ambiguity is alleviated, *e.g.*, $\mathcal{C}_{23}$ might better represent the concept *body* which is partitioned into $\mathcal{P}_2$ and $\mathcal{P}_3$ previously. The overall flowchart of our algorithm is summarized in Algorithm 1.

Algorithm 1 : Fine-tuned part segmentation

Input: A union of original regions $\mathcal{R} = \bigcup_{l=1}^L \mathcal{R}_l$.
1: Choose four seed parts $\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3, \mathcal{P}_4$;
2: $\mathcal{R} \leftarrow \mathcal{R} - \bigcup_{i=1}^4 \mathcal{P}_i$;
3: **REPEAT**: for a region $\mathcal{R}_l \subset \mathcal{R}$;
4: **IF** $\mathcal{R}_l$ is connected to at least one seed parts, let $\mathcal{P}_i$ be the largest one, $i \in \{1, 2, 3, 4\}$;
5: $\mathcal{P}_i \leftarrow \mathcal{P}_i \bigcup \mathcal{R}_l$ and $v_l' \leftarrow v_i$;
6: $\mathcal{R} \leftarrow \mathcal{R} - \mathcal{R}_l$;
7: **ENDIF**
8: **UNTIL**: $\mathcal{R} = \varnothing$ or the remaining regions are isolated.
9: Four refined parts $\{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3, \mathcal{P}_4\}$.
10: $\mathcal{C}_{23} \leftarrow \mathcal{P}_2 \bigcup \mathcal{P}_3$.
Output: Fine-tuned parts $\{\mathcal{P}_1, \mathcal{C}_{23}, \mathcal{P}_4\}$.

4. EXPERIMENTS

4.1. Dataset and Implementation Details

We evaluate our algorithm on the CUB-200-2011 dataset [7], one of the most popular datasets for fine-grained visual categorization. There are 11788 *bird* images among 200 different species, and a bounding box containing exactly one *bird* is provided for each sample. Throughout this paper, we inherit the fixed training/testing split provided by the authors.

For the BoF model, we first resize each image (after cropped by the bounding box) so that the longer axis has 600 pixels. Two types of handcrafted descriptors are extracted, *i.e.*, Max-SIFT [29] and LCS [15]. All the descriptors are reduced to 64 dimensions with PCA. Following [4], we train a GMM with 128 components to encode the descriptors on the foreground, and four GMMs with 32 components, one for each part. Fisher vectors followed by sum-pooling are adopted for encoding, and features from different parts are concatenated into an image-level representation vector.

We also extract deep conv-net features [30] to cooperate with our algorithm. An image, after cropped by the bounding box, is resized to 224×224 and fed into a deep CNN. The intermediate responses referred to as fc-6 are taken as the image representation (4096 dimensions). We use PCA to reduce the vectors extracted from four parts to 1024 dimensions.

Thus, we obtain four types of features, *i.e.*, MSIFT (Max-SIFT), LCS, Fused (MSIFT+LCS) and deep conv-net. We denote feature vectors summarized from the whole foreground and four individual parts as $\mathbf{x}_{\mathrm{FG}}$ and $\mathbf{x}_{\mathrm{PART}}$, respectively. Superscripts O and R denote features extracted from before and after fine-tuned segmentation. All the features are square-root normalized and then ℓ_2 normalized before fed into an SVM.

4.2. Results and Comparisons

The categorization performance is evaluated by **mean accuracy** (mA). Results provided by different features and models are summarized in Table 1. One might observe that the proposed algorithm outperforms the baseline in almost every single case. For example, with MSIFT, the recognition rate is 45.17% after fine-tuned segmentation, with a more than 1.3% absolute gain (a more than 3% relative accuracy gain). The consistent improvement reveals the advantage of fine-tuned segmentation. The performance produced by concatenating

Table 1: Classification accuracy (%) on CUB-200-2011 using different models and features.

	LCS	MSIFT	Fused	Deep Conv-net
$\mathbf{x}_{\mathrm{FG}}^{\mathrm{O}}$	47.82	**46.60**	60.37	68.29
$\mathbf{x}_{\mathrm{FG}}^{\mathrm{R}}$	**47.99**	46.31	**60.66**	**69.83**
$\mathbf{x}_{\mathrm{PART}}^{\mathrm{O}}$	48.24	43.85	58.16	61.54
$\mathbf{x}_{\mathrm{PART}}^{\mathrm{R}}$	**49.51**	**45.17**	**59.55**	**63.00**
$[\mathbf{x}_{\mathrm{FG}}^{\mathrm{O}}; \mathbf{x}_{\mathrm{PART}}^{\mathrm{O}}]$	53.58	50.20	63.29	67.81
$[\mathbf{x}_{\mathrm{FG}}^{\mathrm{R}}; \mathbf{x}_{\mathrm{PART}}^{\mathrm{R}}]$	**54.36**	**51.18**	**64.34**	**69.02**

$\mathbf{x}_{\mathrm{FG}}$ and $\mathbf{x}_{\mathrm{PART}}$ is usually better than that produced by using $\mathbf{x}_{\mathrm{FG}}$ and $\mathbf{x}_{\mathrm{PART}}$ separately. The only exception comes from the result of using deep conv-net features, which might be caused by the relatively weak descriptive power of conv-net features to describe individual parts.

The impact of constructing mid-level parts is summarized in Table 2. In order to keep the feature length unchanged after part combination, we use a GMM with 64 components and reduce the dimension of deep conv-net features from 4096 to 2048 to encode the mid-level parts. With a mid-level part composed of $\mathcal{P}_2$ and $\mathcal{P}_3$, our algorithm achieves higher accuracy. We point out that such improvement comes from alleviating the ambiguity on the boundary of segmented parts. $\mathcal{P}_2$ and $\mathcal{P}_3$, two middle parts, are combined to form the *body* of a *bird*, which might be more robust than individual parts.

In Table 3, we compare our algorithm with the state-of-the-arts. One can observe that our algorithm outperforms the baselines and is also competitive among recently proposed works such as [3] [20]. Since we do not train part detectors using the complicated and time-consuming part-based R-CNN model, our result is inferior to [22].

4.3. Discussions

To reveal that the accuracy gain does come from the fine-tuned segmentation, we perform pairwise classification experiments. We choose 10 *bird* classes with 10 well-segmented cases and 10 poor cases in each class. Then, binary classification is performed between any two classes (45 tasks in total). In each task, we randomly split both well- and poorly-segmented samples into training and testing groups (5 images each). Mean classification accuracy is reported over 45 tasks.

Mean classification accuracy with the original segmentation is 58% on the poorly-segmented samples, and 82% on the well-segmented ones, with a 24% difference. After fine-tuned segmentation, accuracy on the originally poor-segmented samples rises to 72%, with a 14% absolute (24% relative) accuracy gain. In comparison, we also perform a 10-class categorization on these 200 samples. In each class, 10 samples are randomly selected for training and the rest are used for testing. The recognition rate rises from 60% to 70.5% with fine-tuned segmentation. All the above experiments reveal the benefit of fine-tuned segmentation.

Table 2: Classification accuracy (%) using mid-level parts.

	Fused	Deep Conv-net
$[\mathbf{x}_{\mathrm{FG}}; \{\mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_3, \mathcal{P}_4\}]$	64.34	69.02
$[\mathbf{x}_{\mathrm{FG}}; \{\mathcal{C}_{12}, \mathcal{P}_3, \mathcal{P}_4\}]$	62.52	67.00
$[\mathbf{x}_{\mathrm{FG}}; \{\mathcal{P}_1, \mathcal{C}_{23}, \mathcal{P}_4\}]$	**65.13**	**70.34**
$[\mathbf{x}_{\mathrm{FG}}; \{\mathcal{P}_1, \mathcal{P}_2, \mathcal{C}_{34}\}]$	63.64	68.00

Table 3: Classification accuracy (%) comparison with the state-of-the-art results on the CUB-200-2011 dataset.

Approaches	**Accuracy**
Deformable Part Descriptors (DPD) [3]	50.98
Part-based One-vs-One Features (POOF) [31]	56.78
Nonparametric Part Transfer [32]	57.84
Symbiotic model fitting [4]	59.4
Fisher vectors with original parts [6] (baseline)	62.7
Fisher vectors with fine-tuned parts	**65.13**
CNN off-the-shelf [20]	65.0
DPD + DeCAF [3]	64.96
Part-based R-CNNs [22]	**76.37**
Deep conv-net with original parts (baseline)	67.81
Deep conv-net with fine-tuned parts	**70.34**

5. CONCLUSIONS

In this paper, we tackle the problem of over-segmentation, which is frequently observed in the case of unsupervised part detection on fine-grained visual concepts. Based on a straightforward intuition, we perform a greedy algorithm and a bruteforce search to merge and combine smaller parts into large ones. Despite the simplicity, our algorithm provides a significant accuracy gain on the CUB-200-2011 dataset based on unsupervised part alignment [6], and achieves competitive performance with both Fisher vectors and deep conv-net features. By this, we verify that the improvement does come from fine-tuned segmentation. In the future, we will generalize our work to other part detectors such as [4] and other fine-grained datasets such as [10].

6. ACKNOWLEDGEMENTS

This work was supported by the National Natural Science Foundation of China (Grant No. 61402079). We would like to thank Ming Li for helpful discussions about refining method and his comments on the paper.

7. REFERENCES

[1] S. Yang, L. Bo, J. Wang, and L. Shapiro, "Unsupervised template learning for fine-grained object recognition," in *Advances in Neural Information Processing Systems*, 2012, pp. 3131–3139.

[2] B. Yao, G. Bradski, and L. Fei-Fei, "A codebook-free and annotation-free approach for fine-grained image categorization," in *Computer Vision and Pattern Recognition*, 2012, pp. 3466–3473.

[3] N. Zhang, R. Farrell, F. Iandola, and T. Darrell, "Deformable part descriptors for fine-grained recognition and attribute prediction," in *International Conference on Computer Vision*, 2013, pp. 729–736.

[4] Y. Chai, E. Rahtu, V. Lempitsky, L. Van Gool, and A. Zisserman, "Symbiotic segmentation and part localization for fine-grained categorization," in *International Conference on Computer Vision*, 2013, pp. 321–328.

[5] L. Xie, Q. Tian, R. Hong, S. Yan, and B. Zhang, "Hierarchical part matching for fine-grained visual categorization," in *International Conference on Computer Vision*, 2013, pp. 1641–1648.

[6] E. Gavves, B. Fernando, C. Snoek, A. Smeulders, and T. Tuytelaars, "Fine-grained categorization by alignments," in *International Conference on Computer Vision*, 2013, pp. 1713–1720.

[7] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The Caltech-UCSD Birds-200-2011 Dataset," Tech. Rep. CNS-TR-2011-001, California Institute of Technology, 2011.

[8] M. Nilsback and A. Zisserman, "Automated flower classification over a large number of classes," in *Computer Vision, Graphics & Image Processing*, 2008, pp. 722–729.

[9] S.Maji, E.Rahtu, J.Kannala, M.Blaschko, and A.Vedaldi, "Fine-grained visual classification of aircraft," *CoRR*, vol. abs/1306.5151, 2013.

[10] A. Khosla, N. Jayadevaprakash, B. Yao, and L.Fei-Fei, "Novel dataset for fine-grained image categorization," in *First Workshop on Fine-Grained Visual Categorization, Computer Vision and Pattern Recognition*, 2011.

[11] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, pp. 91–110, November 2004.

[12] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition*, 2005, vol. 1, pp. 886–893.

[13] F. Perronnin, J. Snchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *European Conference on Computer Vision*, 2010, pp. 143–156.

[14] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *Computer Vision and Pattern Recognition*, 2010, pp. 3360–3367.

[15] J. Sanchez, F. Perronnin, T. Mensink, and J. Verbeek, "Image classification with the fisher vector: Theory and practice," *International Journal of Computer Vision*, vol. 105, pp. 222–245, December 2013.

[16] L. Xie, Q. Tian, M. Wang, and B. Zhang, "Spatial pooling of heterogeneous features for image classification," in *Transactions on Image Processing*, 2014, vol. 23, pp. 1994–2008.

[17] L. Xie, J. Wang, B. Guo, B. Zhang, and Q. Tian, "Orientational pyramid matching for recognizing indoor scenes," in *Computer Vision and Pattern Recognition*, 2014, pp. 3734–3741.

[18] A. Krizhevsky, S. Ilya, and G. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.

[19] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[20] A.Razavian, H.Azizpour, J.Sullivan, and S.Carlsson, "CNN features off-the-shelf: an astounding baseline for recognition," *CoRR*, vol. abs/1403.6382, 2014.

[21] 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," *CoRR*, vol. abs/1310.1531, 2013.

[22] N. Zhang, J. Donahue, R. Girshick, and T. Darrell, "Part-based r-cnns for fine-grained category detection," in *European Conference on Computer Vision*, 2014, pp. 834–849.

[23] L. Xie, Q. Tian, R. Hong, and B. Zhang, "Image classification and retrieval are one," in *International Conference on Multimedia Retrieval*, 2015.

[24] Y. Chai, E. Rahtu, V. Lempitsky, L. Van Gool, and A. Zisserman, "Tricos: A tri-level class-discriminative co-segmentation method for image classification," in *European Conference on Computer Vision*, 2012, pp. 794–807.

[25] R. Farrell, O. Oza, N. Zhang, V. Morariu, T. Darrell, and L. Davis, "Birdlets: Subordinate categorization using volumetric primitives and pose-normalized appearance," in *International Conference on Computer Vision*, 2011, pp. 161–168.

[26] N. Zhang, R. Farrell, and T. Darrell, "Pose pooling kernels for sub-category recognition," in *Computer Vision and Pattern Recognition*, 2012, pp. 3665–3672.

[27] J. Liu, A. Kanazawa, D. Jacobs, and P. Belhumeur, "Dog breed classification using part localization," in *European Conference on Computer Vision*, 2012, pp. 172–185.

[28] X. Zhang, H. Xiong, W. Zhou, and Q. Tian, "Fused one-vs-all mid-level features for fine-grained visual categorization," in *Proceedings of the ACM International Conference on Multimedia*, 2014, pp. 287–296.

[29] L. Xie, Q. Tian, and B. Zhang, "Max-sift: Flipping invariant descriptors for web logo search," in *International Conference on Image Processing*, 2014, pp. 5716–5720.

[30] A. Vedaldi and K. Lenc, "Matconvnet – convolutional neural networks for matlab," *CoRR*, vol. abs/1412.4564, 2014.

[31] T. Berg and P. Belhumeur, "Poof: Part-based one-vs.-one features for fine-grained categorization, face verification, and attribute estimation," in *Computer Vision and Pattern Recognition*, 2013, pp. 955–962.

[32] G. Christoph, R. Erik, F. Alexander, and D. Joachim, "Non-parametric part transfer for fine-grained recognition," in *Computer Vision and Pattern Recogniti*, 2014, pp. 2489–2496.