MINE THE FINE: FINE-GRAINED FRAGMENT DISCOVERY

M. Hadi Kiapour

University of North Carolina at Chapel Hill
Chapel Hill, NC, USA
hadi@cs.unc.edu

Wei Di, Vignesh Jagadeesh, Robinson Piramuthu

eBay Research Labs, San Jose, CA, USA
{wedi, vjagadeesh, rpiramuthu}@ebay.com

ABSTRACT

While discriminative visual element mining has been introduced before, in this paper we present an approach that requires minimal annotation in both training and test time. Given only a bounding box localization of the foreground objects, our approach automatically transforms the input images into a roughly-aligned pose space and discovers the most discriminative visual fragments for each category. These fragments are then used to learn robust classifiers that discriminate between very similar categories under challenging conditions such as large variations in pose or habitats. The minimal required input, is a critical characteristic that enables our approach to generalize over visual domains where expert knowledge is not readily available. Moreover, our approach takes advantage of deep networks that are targeted towards fine-grained classification. It learns mid-level representations that are specific to a category and generalize well across the category instances at the same time. Our evaluations demonstrate that the automatically learned representation based on discriminative fragments, significantly outperforms globally extracted deep features in classification accuracy.

Index Terms— Fine-grained, mid-level representation, deep learning, classification

1. INTRODUCTION

Fine-grained recognition takes the problem of generic object categorization to the next level where the goal is to discriminate between categories of very similar appearance, e.g. bird species [1]. Due to the subtle differences between the subordinate categories, several works have relied on domain experts for detailed labeling of discriminative attributes [2] or key point locations [3, 4, 5, 6, 7]. Typically, human-labeled key-point annotations are used to identify the object pose or perform some kind of alignment as a pre-processing step [4, 6, 3, 8]. There are two major disadvantages in relying on human labeling. First, manual annotation acquisition can be extremely expensive and requires strong domain knowledge. Second, while animals or airplanes have well-defined key points, many other categories (e.g. food) lack such precisely defined feature points. Hence, automatic deriving approaches for fine-grained categorization have drawn a lot of attention recently [9, 10, 11, 12, 13, 8]. Localizing discriminative elements of each category remains the main bottle-neck of fine-grained classification approaches. Our approach requires minimal annotation in both training and test time and automatically discovers the most discriminative fragments of each category. In particular, we propose an alignment-based method that *only requires a bounding box* which roughly contains the foreground object. This alignment is then used for extracting local discriminative fragments and creating mid-level image descriptors.

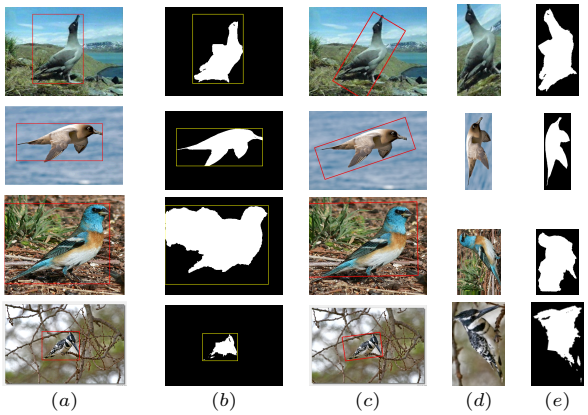

Fig. 1. (a) Original (b) GrabCut Mask (c) Best Rectangle (d) Oriented rectangle (e) Oriented mask. Note that often, even for bad mask, the alignment by the best oriented rectangle is acceptable.

Identifying discriminative mid-level visual elements has recently attracted a lot of attention [14, 15, 13, 16]. Inspired by the visual mining of discriminative blocks for scene classification [16], which uses HOG features to find and encode groups of visually similar patterns, we explore a large set of potentially discriminative regions. However, unlike [16] that uses hand-crafted features to categorize generic categories, we *learn high-level representations* to build robust mid-level visual models. These models are trained to detect a specific discriminative pattern *within a category*.

Recent success of deep learning methods in image clas-

sification [17] and object detection [18], suggests the high capacity of convolutional neural networks for many computer vision problems such as fine-grained classification. Recently, Branson *et al.* [6] used deep activations in a pose aligned space of bird categories and achieved a significant boost in classification and part localization accuracy. Girshick *et al.* [5] proposed a joint object detection and part localization system that deploys deep features along with domain-specific geometric priors to localize parts. However in both of these methods, part annotations are required for training part detectors. Our approach takes advantage of the powerful representation provided by deep networks that are fine-tuned to our specific task, without relying on any sort of part annotation.

The rest of the paper is organized as follows: Section 2 describes all the steps in our approach: alignment, mining fragment proposals, feature extraction, building and ranking fragment sets. Next we introduce our image-level representation based on the discovered discriminative fragments. Section 3 explains the experimental framework and discusses the effect of various factors including number of discriminative fragments and fine-tuning deep networks. Finally, Section 4 concludes with directions for future work.

2. MINE THE FINE

2.1. Discovering Fine-Grained Fragments

2.1.1. Semi-Supervised Alignment

Our method builds on top of the popular GrabCut segmentation [19] to roughly align the foreground objects. To extract the foreground mask, we assume that a bounding box containing the full extent of the foreground object is given. We initialize the GrabCut by setting the area outside of the box as background and inside as probably foreground. We also set the center of the bounding box as foreground which leads to considerably better foreground masks which in return help in the ultimate goal of fine-grained classification. Next, we locate the convex hull of the output foreground mask and fit a rotated rectangle of the minimum area enclosing the hull. We call this rectangle, the *aligned bounding box*. An aligned bounding box is constrained to enclose the foreground region as tightly as possible and also has degrees of freedom to rotate and scale to roughly align with the pose of the bird. There are three major advantages in using aligned bounding boxes. First, they bring the focus to the relevant regions for fine-grained classification (foreground) by reducing the effects of background significantly. Secondly, they allow us to transform the original image space of birds with huge variations in aspect and orientation to a more pose-consistent space. This space allows more comparability for corresponding locations in bounding boxes. Lastly, they can be computed very fast. We call our alignment framework *semi-supervised* as it does not require the ground-truth or detected part locations. Figure 1 shows examples of aligned bounding boxes.

Fig. 2. Oriole: Examples of top mined fragments with the least area under the entropy-rank curves.

2.1.2. Generating Fragment Proposals

We use selective search [20] to generate a large set of region proposals that are potentially discriminative. Although our approach can be used with any method of generating candidate regions, using selective search has a major benefit: It can circumvent the need for searching over the space of all possible positions, scales and aspect ratios. Selective search computes multiple hierarchical graph-based image segmentation [21] over different color spaces and returns a set of bounding boxes to which we refer as *fragments*. We first augment the training set with horizontally flipped images and use "fast" diversification strategy of selective search, i.e. two color spaces: HSV and Lab and two similarity measures: ColorTextureSizeFill and TextureSizeFill and prune out any region with a side length less than 30 pixels. This set of fragments is diverse and provides good coverage across object instances. For efficiency, we select a random subset of the extracted fragments and measure their discriminative power in the next steps.

2.1.3. Feature Extraction

We use the prevalent deep learning tool, Caffe [22] with the architecture of Krizhevsky *et al.* [17] pre-trained on ImageNet which achieved state-of-the-art performance in ILSVRC 2012 classification challenge. Additionally, in order to improve the network's discriminative capability in our specific fine-grained classification, we fine-tune the model to classify the bird categories in the our dataset. In particular, we replace the last layer of 1000 units with a new layer of 200 units, one for each bird category. We continue tuning for 500 iterations with the base learning rate, momentum and weight decay as 0.001, 0.9 and 0.0005 respectively. Throughout the paper, we represent every input image as the activations of the fully connected layer fc-6 (4096 dimensions). Recent study by Branson *et al.* [6] indicates that the later fully connected layers of this CNN architecture significantly outperform earlier lay-

Table 1. Average group accuracy before and after fine-tuning the CNN for selected groups

Group	Indices	fg bbox	fg bbox + vert. fragments	Finetuned CNN?	unnorm. max	norm. max	fg bbox + unnorm. max	fg bbox + norm. max	Top 50 Fragments
Gull	61:66	68.24	62.35	N	77.06	76.47	78.24	**80.59**	-
				Y	71.76	77.65	70.59	80.00	**100.00**
Kingfisher	79:83	84.67	86.67	N	91.33	92.00	92.00	**94.67**	-
				Y	92.00	93.33	92.00	93.33	**100.00**
Oriole	95:98	75.63	86.55	N	90.76	93.28	91.60	**95.80**	-
				Y	93.28	93.28	94.12	**97.48**	96.52
Sparrow	113:133	53.67	55.37	N	70.62	74.58	72.88	**75.71**	-
				Y	83.05	82.49	80.23	81.92	**100.00**
Swallow	135:138	65.83	73.33	N	89.17	**91.67**	89.17	**91.67**	-
				Y	**96.67**	**96.67**	95.83	95.83	77.78
Tern	141:147	43.54	48.80	N	66.03	67.94	62.20	**68.90**	-
				Y	71.77	73.68	72.73	74.64	**98.77**
Vireo	151:157	59.30	60.80	N	71.86	72.86	73.37	**73.87**	-
				Y	77.39	**78.39**	**78.39**	77.39	72.22
Warbler	158:182	66.89	66.89	N	-	-	-	-	-
				Y	69.73	68.78	69.59	71.22	**98.78**
Woodpecker	187:192	94.67	95.27	N	92.31	91.72	92.31	**95.86**	-
				Y	93.49	94.08	93.49	**96.45**	95.83
Wren	193:199	68.57	60.95	N	70.95	67.62	70.48	**71.43**	-
				Y	73.81	73.81	72.86	76.67	**99.17**
Average		68.10	69.70	N	72.01	72.81	72.22	**74.85**	-
				Y	82.29	83.22	81.98	84.49	**93.91**

ers in classification. In order to extract the deep activations, each input image is warped to 256×256 and central crop of size 227×227 is picked as the fixed-sized input to the first layer. In order to account for the effects of warping and cropping, prior to feature calculation, every input fragment is extracted at a larger height and width proportional to its size such that the final central patch used for feature calculation, exactly corresponds the the originally extracted fragment.

2.1.4. Building Fragment Sets

Having a large pool of fragments in hand, our goal is to learn visual models that can 1) detect highly discriminative regions across categories and 2) generalize over different instances within a category. We tackle this problem by generating training sets formed within categories: Starting from every fragment we iteratively expand the set by adding more fragments from *distinct* training instances of the same category. This is a critical step in order to assure generalization. Given a set of n categories, for each of the extracted fragments in the training set, we train a one-vs-all classifier that discriminates between the category from which the fragments are extracted and all other $n-1$ categories. We iteratively refine the trained model in two steps: 1) We train the model on all fragments in the set including the newly added training fragments and 2) we apply the model to other candidate fragments, sort them based on the confidence of belonging to the same category and add the top m scoring fragments as new training samples to the set. In this process, we enforce two constrains: First, we make sure that the new added fragments do not already exist in the training set and 2) each of the new fragments must be from a different image. These constrains ensure that the trained model does not overfit to learning a particular instance of the category and guarantees to increase diversity in each iteration. We continue this process for t iterations. We heuristically set m and t to be 10 and 5 respectively.

In order to accelerate the learning process, we use the efficient LDA classifiers with closed-form updates which bypasses the need for extensive hard-negative mining [23]. In particular, given a set of n target categories, we need to compute the sample mean μ_- of the negative examples and sample covariance matrix S of the entire training set only once. For a binary LDA we assume the classes have a shared covariance matrix S, and only need to update μ_+ in each iteration. The resulting LDA classifier for the positive class is obtained as follows [24]:

$$w \propto S^{-1}(\mu_+ - \mu_-) \tag{1}$$

2.1.5. Ranking Fragment Sets

In order to identify the most discriminative sets, we adopt the measure based on Entropy-Rank curve introduced in [16]. We take the final models trained on all fragment sets and perform a binary classification on the fragments in the validation set. The fragments are sorted based on their score, and the top k ranking ones are selected. Then the entropy $H(y|k)$ is computed as follows:

$$H(Y|k) = \sum_{y=1}^{n} p(y|k) \log_2 p(y|k) \tag{2}$$

Where n is the number of target categories and $p(y = y_i|k)$ is the fraction of the top scoring k fragments that have the label y_i.

Next, we compute the area under the curve (AUC) of the entropy-rank curves which is analogous to average precision calculated from Precision-Recall curves. For an ideal classifier, the entropy starts at zero and remains zero up to a very high number of retrieved fragments. It then starts to increase due to the fragments that are returned from classes other than the target class. For each bird category, we pick the r fragment sets with the lowest AUC values. Therefore, for all n categories we obtain $n \times r$ detectors of the most discrimina-

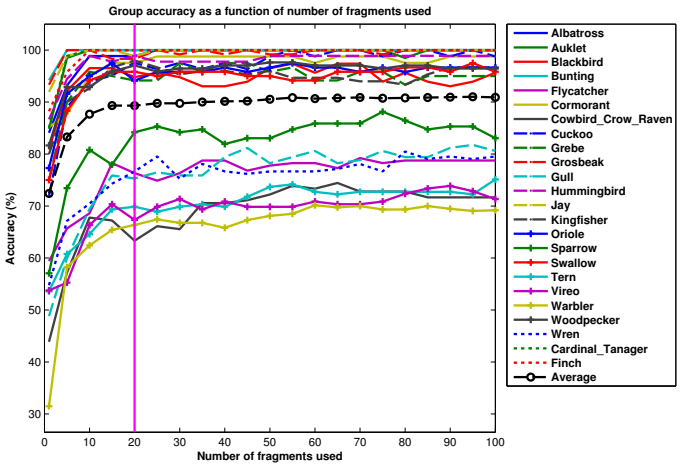

Fig. 3. Classification accuracy vs. number of parts.

tive visual elements in total. Some examples of fine-mined fragments are also shown in Figure 2.

2.2. Representation and Classification

Given an image, a set of fragments is extracted as described in 2.1.2 with their descriptors as in 2.1.3. We apply each detector to all fragments of the image and concatenate the maximum scores found to build our image-level descriptor. In order to build more robust features, we also add an optional normalization step to our max-pooled fragment descriptors which maps the feature vector to real values in $[0, 1]$. Finally a n-class SVM classifier is trained using liblinear [25].

3. EXPERIMENTAL RESULTS

3.1. Experimental Framework

We evaluate our approach on the CUB-200-2011 [1] dataset consisting of 200 bird categories. A wide range of challenges for categorization are presented in this dataset, from pose variations and various bird habitats to very similar species (e.g. 25 kinds of warblers, 21 types of sparrows, etc.). Birds within each group share higher similarity and are more difficult to discriminate. Therefore, we form a new setting, in which we focus on building models to classify within 24 manually defined groups of similar species, grouped based their name convention and visual similarity. Table 1 lists 10 selected groups. We have included the numerical indices of the merged categories in CUB dataset. We use the standard train/test split and the provided bounding boxes in the dataset.

3.2. Comparing the Proposed Method with Baseline

Baseline 1: FG-BBOX: We use the provided foreground bounding box information to extract deep features. The feature dimension is 4096.

Baseline 2: FG-BBOX + Ver. Fragments: We further divide the aligned bounding box area into 4 vertical regions and concatenate their CNN features with the foreground bbox. The final dimension is $4096 + 4 * 4096$. Table 1 shows the results on the selected groups compared to baseline using pre-trained and fine-tuned CNN features. The last column lists the results by just using the top 50 fragment detectors. The proposed method significantly outperforms the baseline. Particularly, we notice that for most of the *hard* groups, e.g. gull, oriole, etc., the proposed method boosts accuracy between 10% to 20%. For sparrow and wabbler, we still gain less significant improvement (1% to 3%). One exception is the *easy* group woodpecker, where we achieved similar accuracy. This makes sense as improving an already well-performing classification is much harder.

Advantage of using FG Bounding Box: Adding FG bbox helps the max pooled feature. This is mainly due to the complimentary nature of CNN feature extracted from the FG (global structure, shape, color) and fragments (local details). Also FG CNN feature helps to bring some useful context information when necessary, e.g. the birds habitat.

Fine-tuning: We gain slight improvements by using the fine-tuned CNN features. This is possibly due to the fact that the fine-tuning is conducted over 200-classes rather than species in each group. However, fine-tuned features shows more improvements for *hard* groups as compared to *easy* groups, for example, tern vs. woodpecker.

Different Number of Fragments: Figure 2.2 shows the classification accuracy vs. the number of top fragment detectors. The x-axis shows the number of selected top detectors for each class within the group, varying from 5 to 100. Some examples of fine-mined fragments are shown in Figure 2. Figure 2.2 also plots the averaged accuracy over all 24 groups. It can be seen from Figure 2.2 that classification accuracy increases very fast in the beginning, and becomes stabilized around 20 fragment-detector. This means that using quite a few fragment detectors, our proposed algorithm can already achieve good results. Given few detectors, the final feature dimension of the proposed algorithm is also much less than the baseline approach. Comparing results using just the top 50 fragment detectors with other results in Table 1, which do not sort nor select the best fragment detectors, we found the results are better using just the top detectors. This is reasonable as some of the detectors may not have valuable or discriminated information, and including them can introduce noisy values into the representation.

4. CONCLUSIONS

We proposed a fine-grained categorization framework that only relies on weak foreground localizations to align and discover the discriminative elements of each category. Empowering mid-level representations with task-specific learned representations, we can outperform the current CNN feature baselines. Future work includes reducing redundancy in the visual models and improving candidate fragments generation.

5. REFERENCES

[1] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The caltech-ucsd birds-200-2011 dataset.," *Technical Report*, 2011.

[2] C. Wah, S. Branson, P. Perona, and Belongie S., "Multiclass recognition part locatization with humans in the loop," *ICCV*, 2011.

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

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

[5] Ning Zhang, Jeff Donahue, Ross Girshick, and Trevor Darrell, "Part-based r-cnns for fine-grained category detection," *ECCV*, 2014.

[6] Steve Branson, Grant Van Horn, Serge Belongie, and Pietro Perona, "Bird species categorization using pose normalized deep convolutional nets," *ECCV*, 2014.

[7] N. Zhang, R. Farrell, and T. Darrell, "Pose pooling kernels for sub-category recognition," *CVPR*, 2012.

[8] Efstratios Gavves, Fernando Basura, Cees G.M. Snoek, Arnold W.M. Smeulders, and Tinne Tuytelaars, "Local alignments for fine-grained categorization," *IJCV*, 2014.

[9] G. Martínez-Muñoz and et al., "Dictionary-free categorization of very similar objects," *CVPR*, 2009.

[10] Shulin Yang, Liefeng Bo, Jue Wang, and Linda G. Shapiro, "Unsupervised template learning for fine-grained object recognition," *NIPS*, 2012.

[11] Bangpeng Yao, Aditya Khosla, and Li Fei-Fei, "Combining randomization and discrimination for fine-grained image categorization," *CVPR*, 2011.

[12] Yao Bangpeng, Gray Bradski, and Li Fei-Fei, "A codebook and annotation-free approach for fine-grained image categorization," *CVPR*, 2012.

[13] Y. J. Lee, A. A. Efros, and M. Hebert, "Style-aware mid-level representation for discovering visual connections in space and time," *ICCV*, 2013.

[14] C. Doersch, S. Singh, A. Gupta, J. Sivic, and A.A Efros, "What makes paris look like paris?," *ACM Transactions on Graphics (SIGGRAPH)*, 2012.

[15] S. Singh, A. Gupta, and A. Efros, "Unsupervised discovery of mid-level discriminative patches," *ECCV*, 2012.

[16] Juneja M., Vedaldi A., Jawahar C. V., and Zisserman A., "Blocks that shout: Distinctive parts for scene classification," *CVPR*, 2013.

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

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

[19] Kolmogorov V. Rother, C. and A. Blake, "Grabcut: Interactive foreground extraction using iterated graph cuts," *ACM Trans. On Graphics*, 2004.

[20] J. Uijlings, van de Sande, K. T. Gevers, and Smeulders. A. S., "Selective search for object recognition," *IJCV*, 2013.

[21] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *IJCV*, 2004.

[22] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell, "Caffe: Convolutional architecture for fast feature embedding," *arXiv preprint arXiv:1408.5093*, 2014.

[23] B. Hariharan, J. Malik, and D. Ramanan, "Discriminative decorrelation for clustering and classification," *ECCV*, 2012.

[24] C. M. Bishop, "Pattern recognition and machine learning," *Springer New York*, 2006.

[25] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "Liblinear: A library for large linear classification," *Journal of Machine Learning Research*, 2008.