Fast Bird Part Localization for Fine-Grained Categorization

Yaser Souri
Sharif University of Technology
ysouri@ce.sharif.edu

Shohreh Kasaei
Sharif University of Technology
skasaei@sharif.edu

Abstract

Incorporating precise part information has proved to be crucial in building accurate fine-grained categorization systems in recent studies. The state-of-the-art approach for part localization uses a convolutional neural network and needs thousands of forward passes of the network, which is very time consuming. In this paper, an efficient method is proposed for part localization, with only one forward pass of the network. The proposed method provides improved generalization capability, compared to the state-of-the-art, and the ability to detect multiple part instances simultaneously, without much computational overhead. Experiments on the Caltech-UCSD Birds dataset show that the proposed method, while being much faster, achieves comparable accuracy to the state-of-the-art.

1. Introduction

Many recent papers on fine-grained categorization reveal the importance of accurate part location information. For instance, the methods proposed in [1] and [8] are able to significantly boost their categorization performance using ground truth part locations at test time compared to when they estimate the part locations. Since the availability of part locations at test time is unrealistic, it is important to precisely localize parts automatically for build an accurate fine-grained categorization system.

The state-of-the-art method of PartRCNN [8] uses the RCNN [3] approach for localizing parts. RCNN while producing excellent results for object detection, and in this case part localization, needs approximately 2000 forward passes of a typical *convolutional neural network* (CNN), which is very time consuming. This huge computational overhead makes PartRCNN unpractical for real-time and low power use cases (e.g., mobile devices). In this work, an efficient method that can be used for part localization (e.g., localizing a bird-head) is introduced. Compared to PartRCNN, our method has many desirable properties: (i) *Computational complexity*. It only needs one forward pass of the same typical CNN, which makes it orders of magnitude faster than PartRCNN. (ii) *Generalization*. This method can localize parts of objects from unseen classes, in difficult poses, and even on pictures of pencil drawings. It can also be trained to localize the bounding box of the whole object. (iii) *Multiple part localization*. This method is able to localize multiple part instances without much computational overhead (still using only one forward pass of the CNN).

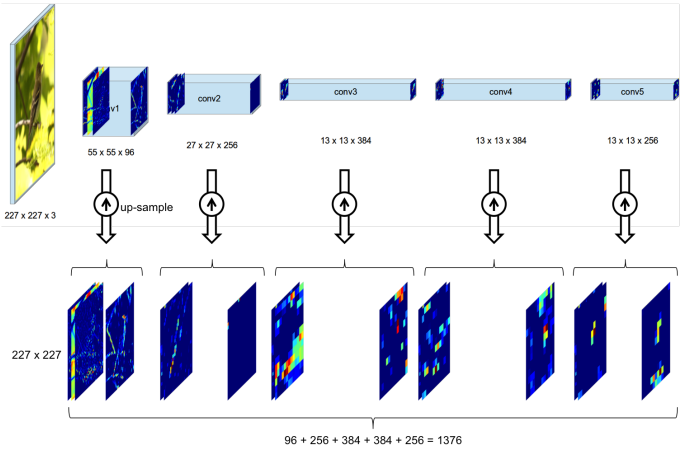

Figure 1. Overview of feature extraction step using the AlexNet [5] architecture. Input image is resized to a fixed dimension. CNN computes the feature maps in convolutional layers. Each feature map is then up-sampled to the fixed input dimension.

2. Proposed Method

We cast the problem of bird part localization as classifying pixels of the image to whether each pixel belongs to the part (positive) or not (negative). To this end, a feature vector for each pixel and a classifier for each part is needed. The convolutional layers of AlexNet [5] network pretrained on ILSVRC dataset [6] is used to extract the feature vectors. First, the input image is resized to the input dimension of AlexNet (i.e., 227x227). Then, with a single forward pass, the convolutional feature maps (*conv1-conv5*) for the input image is computed. Each feature map is then up-sampled to 227x227 as shown in Figure 1.

For AlexNet architecture, this gives a total of 1376 fea-

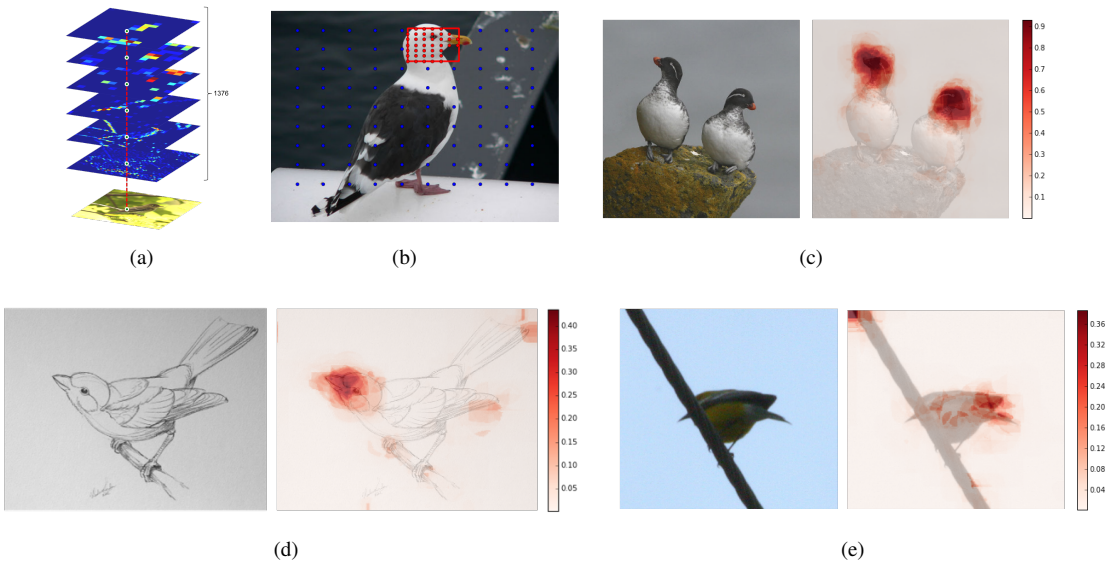

Figure 2. (a) Feature vector for each pixel is extracted from up-sampled feature maps. (b) A set of positive (red) and negative (blue) points used for training a bird-head localizer. In (c), (d), and (e) on the left is the input image and on the right is the estimated probability of presence of bird's head-part in each pixel (in red). Colorbar shows the actual probability value estimated for the corresponding image.

ture maps. The feature maps are stacked to form a hypercolumn (inspired by [4]). For each pixel in position (x, y) in the image, a 1376 dimensional feature vector is extracted from this hypercolumn by concatenating filter response values at position (x, y) from all feature maps (Figure 2(a)).

We use a *Random Forest* classifier due to its many desirable properties [2] including but not limited to its speed and its ability to produce probability estimates. For training the classifier, the training set of CUB [7] dataset is used to create two pools of positive and negative pixels with their corresponding pixel-wise deep features. For each image, first the hypercolumn is created with a forward pass of the CNN. Then, a set of 20 positive pixels, uniformly spaced, which are located inside the bounding box of the part (e.g., bird-head) are added to the positive pool of pixels with their corresponding feature values. The same is done for 40 negative points which are not located inside the part bounding box (Figure 2(b) shows these points for an image). Then, the Random Forest is trained to classify positive and negative points discriminatively. At test time, all pixels are (densely) classified using the trained Random Forest resulting in a single channel image where each pixel's value is the probability of that pixel belonging to the part (see Figures 2(c)-(e).) This method can detect multiple parts simultaneously (Figure 2(c)); detect head of birds in pencil drawings (Figure 2(d)) and also detect situations where the head is not clearly visible (Figure 2(e)) with low probabilities.

Classification Pipeline: A similar pipeline as PartRCNN [8] is used for classification. A part bounding box is obtained from each probability estimate of part location (produced by our method) by thresholding and selecting the enclosing rectangle of the largest connected component. Three Random Forests are trained to localize the bird's bounding box, head, and body. Furthermore, for feature extraction (fc7 layer), four different AlexNet networks are fine-tuned (namely, the entire image and the bird's bounding box, body, and head). A linear SVM is used for categorization. With this setup, the method achieves 72.35% mean accuracy for categorization, which is comparable to PartRCNN [8] (73.89%), while being at least 2 orders of magnitude faster.

Acknowledgments: We would like to thank Dr. Ehsan Adeli for his help in writing the paper.

References

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

[2] A. Criminisi, J. Shotton, and E. Konukoglu. Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning. *Foundations and Trends in Computer Graphics and Vision: Vol. 7: No 2-3, pp 81-227*, 2012.

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

[4] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Hypercolumns for object segmentation and fine-grained localization. In *CVPR*, 2015.

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

[6] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. Berg, and L. Fei-Fei. Imagenet large scale visual recognition challenge. *IJCV*, 2015.

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

[8] N. Zhang, J. Donahue, R. Girshick, and T. Darrell. Part-based r-cnns for fine-grained category detection. In *ECCV*, 2014.