Learning Regions and Descriptors for Fine-grained Recognition

Dequan Wang[1], Tianjun Xiao[2], Zhiqiang Shen[1], and Xiangyang Xue[1]

[1]Shanghai Key Laboratory of Intelligent Information Processing, Fudan University
[2]Institute of Computer Science and Technology, Peking University

[1]{dqwang12; zhiqiangshen13; xyxue}@fudan.edu.cn [2]xiaotianjun@pku.edu.cn

Abstract

Fine-grained categorization, which aims to distinguish subordinate-level categories such as bird species or dog breeds, is an extremely challenging task due to two main issues: how to localize discriminative regions for recognition and how to learn sophisticated features for representation. In this paper, we develop a joint representation learning framework which simultaneously detects informative regions and distinguishes subtle differences for subordinate-level categories. The region detectors are learned in unsupervised settings, based on the observation that neural networks for fine-grained recognition have special spatial distributions for regions of interest from object-level to part-level. The appearance descriptor are the concatenation of hierarchical convolutional neural network features encoding both coarse-grained and fine-grained visual differences. Only image-level labels are necessary for training in our approach, which avoids using labor-intensive bounding box or part annotations from end-to-end. Experimental results on challenging fine-grained image dataset demonstrate that despite of the weakest supervision our approach outperforms most of state-of-the-art methods and even achieves accuracy comparable with the methods which heavily rely on extra annotations.

1. Background

Compared to basic-level categorization, it is necessary for subordinate-level classification to explicitly discriminate subtle differences between highly similar subcategories. Progress in fine-grained categorization not only boosts the performance of generic object recognition, but also benefits human beings in some specific domains, while even experts may find it a great trouble to differentiate such subcategories.

2. Method

In general, fine-grained categorization is extremely challenging. This is due to two main issues: how to 1) localize discriminative regions and 2) learn the corresponding feature representations. In this work, we simultaneously tackle both region discovery and representation learning. The key idea is to localize important parts with weakest supervision and to describe subtle difference among the species while discarding useless information for classification. Specifically we present a novel framework utilizing multi-scale regions of interest on convolutional neural network, bypass time-cosing annotation like bounding box or part key point completely. In recognition scenarios, we firstly localize informative regions then capture their subtle visual differences using the learned hierarchy convolutional neural network features, leading to a joint representation for fine-grained recognition.

2.1. Region Discovery

Saliency Heatmap We take advantage of the architecture of VGGNet[1] for training CNNs. Afterward, we remove fully connected layers from the whole network, and only use the last pooling layer to obtain 512 channels of filter response maps. Since filter parameters of CNNs are learned from domain-specific training data, image-level CNNs deliver heat map of spatial distribution of regions of interest (ROIs). Our goal is to detect the saliency in the hidden layers to guide the selection of regions of interest.

We observe that some neurons at the last pooling layer characterize the distributed attributes of objects, which can be seen as texture descriptors or part detectors, while others may catch cluster irrelevant noise, which are actually useless for fine-grained classification. In general, the filter response map shows consistent correspondence when neurons are concerned with the specific domain. For each filter response map of images, we calculate *standard deviation of*

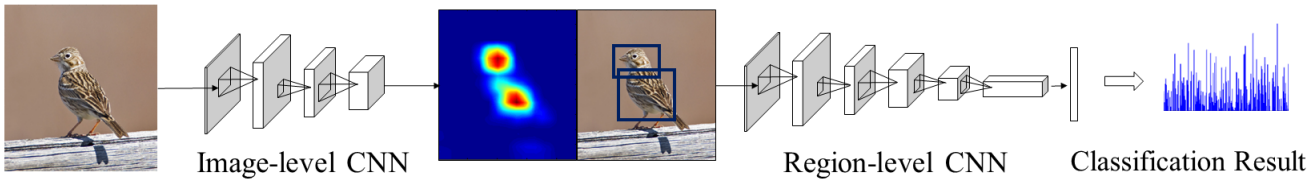

Figure 1. Overview of our framework.

kurtosis (SDK) as follows:

$$SDK_i = \sigma([Kurt(\phi_i(\boldsymbol{I}_k))]_{k=1...N}) \quad (1)$$

where $\phi_i(\boldsymbol{I})$ denotes the i-th filter response map of image $\boldsymbol{I}$, N is the number of images, $\sigma(\cdot)$ is the standard deviation, and $Kurt[\cdot]$ is the *kurtosis* measuring the peakedness of the probability distribution of all responses in the map ϕ_i. The *kurtosis* captures the saliency of the image i-th filter response map for fine-grained task, while the standard deviation of kurtosis is used to choose principal response maps. Thus, to select relevant filter response maps, we design selected filter response map $\psi(\cdot)$ as follows:

$$\psi_i(\boldsymbol{I}) = \begin{cases} \boldsymbol{N}(\phi_i(\boldsymbol{I})) & \text{if } SDK_i > \theta, \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

where θ is a threshold. We then generate heatmap of input image$\boldsymbol{I}$ by summing up all selected response maps $\psi(\boldsymbol{I})$.

Region of Interest Heatmap illustrates energy distribution of interest, and we define concept *energy* for region hypothesis, which is calculated by summing up all the including elements of heatmap. Under the guidance of heatmap, the next step is to filter the irrelevant patches proposed by bottom-up mechanism adaptively. In order to encode the confidence of discriminative regions, we calculate the density of box's energy as follows:

$$\rho(region) = \frac{Energy(region)}{Area(region)} \quad (3)$$

2.2. Feature Representation

Two-Step Finetuning Image-level CNN provides spatial distribution of ROIs, and heatmap helps filter irrelevant background patches. Given the set of regions assigned to input images, a detection work ranks them according to score function $\rho(\cdot)$, and picks positive and negative samples according to two thresholds. They are now fed to the region-level CNN which is initialized from the image-level CNN.

Classification The progression through the region-level network can be seen as a movement from low to mid to high-level features. The pooling layers aggregate plenty of sophisticated structural information with max-pooling operation grab hold of deformable parts, while later fully connected layers summarize complex co-occurrence statistics and drop evidence of spatial location. To handle multiple layers with different scales of magnitude, each representation is normalized independently. Our final feature space concatenates from both pooling layers and fully connected layers and we employ a linear SVM to learn weights for the classification.

3. Results

This section presents performance evaluations on challenging datasets CUB-200-2011[2]. We follow the standard evaluation procedure that no extra annotations except class labels are used in training and report classification accuracy which indicates the average over all test samples.

Analysis of Region Discovery The difference between results using the same number of grained pipeline but with or without bounding box is caused by region detection accuracy. The gap is not significant, but meaningful.

Analysis of Two-Step Finetuning Once a region is discovered, how its associated features are extracted makes a difference. We could feed it into the image-level CNN to extract features. Or, alternatively let the image-level CNN act as region of interest generator which picks up potential region hypothesis according to feature map scores. The selected domain relative patches are used to train the region-level CNN. Results show that, for CUB-200-2011 dataset, this brings the most important gain.

Methods	Annotation	Accuracy (%)
VGG-19[1]	BBox	73.2
Image-level	BBox	76.4
Region-level	BBox	80.5
VGG-19[1]	None	67.0
Image-level	None	75.3
Region-level	None	78.4

Table 1. Evaluation of individual component contributing to the overall performance on CUB-200-2011 dataset[2].

References

[1] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 1, 2

[2] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The caltech-ucsd birds-200-2011 dataset. 2011. 2