A Classify-Before-Detect Method For Weakly Supervised Object Localization

Xiaohu Dong\textsuperscript{1,}\textsuperscript{*}, Ping Wang\textsuperscript{1,b}, Ruigang Fu\textsuperscript{1,c}, Yinghui Gao\textsuperscript{1,d}, Jia Wang\textsuperscript{1,e}

\textsuperscript{1}National Key Laboratory of Science and Technology on ATR, College of Electronic Science, National University of Defense Technology, Changsha, China

\textsuperscript{*}Corresponding author e-mail: xhdong@whu.edu.cn
\textsuperscript{b}760521407@qq.com, \textsuperscript{c}furuiang08@nudt.edu.cn,
\textsuperscript{d}yhgao@nudt.edu.cn, \textsuperscript{e}826216287@qq.com

Abstract. Classify-Before-Detect (CBD) methods provide a reliable initialization for the object detector, and are widely used in weakly supervised object localization (WSOL). This paper proposes a new CBD method for WSOL based on saliency maps. A CNN is first trained to determine the existence of objects in an image. Then, with the trained CNN model for classification, a gradient-based saliency map is obtained to generate candidate locations of objects in the image. Finally, the accurate location coordinates of objects are obtained using the special characteristics of these objects. Experiments have been conducted on the Nexar traffic light dataset, it is shown that CNNs are able to successfully classify images even though the objects occupy only a few pixels in the training images and gradient-based saliency maps provide strong resistant capability to interference. More importantly, our proposed method can locate small objects precisely, which is very difficult for the current object detectors. Moreover, our proposed method is weakly supervised and effective.

1 Introduction

The task of object localization is to identify the location of all instances in an image. Great progress has been made in object localization over the past few years\cite{1}. However, most of the object detectors \cite{2}\cite{3}\cite{4}\cite{5}\cite{6} require strong supervisions in the form of instance-level annotations (e.g. object bounding boxes) which are labor intensive, and these detectors have a bad performance in small objects’ detection. In contrast, weakly supervised object localization (WSOL) learns to localize objects within images using only image-level labels which indicate the presence or absence of an object category in the image. Because image-level labels are much easier to obtain, WSOL has attracted increasingly more interest.

Most existing WSOL methods follow a Detect-Before-Classify (DBC) framework which first generates region proposals, and then determines whether there is an object existing in each region proposal. In contrast, Classify-Before-Detect (CBD) methods first perform whole-image classification, and then utilize its byproduct, the saliency map, to assist detection. CBD methods can be more efficient than DBC methods, since the classification results reduce the search space of detection.

In this paper, we are only interested in single-label datasets (each image in the datasets only has one label). Specifically, the images having at least one target object are labeled as positives. Otherwise, they are labeled as negatives. In this case, we propose an efficient CBD method for Weakly Supervised Object Localization, which localizes small objects (e.g. traffic lights) using a classification CNN without generating region proposals. Our work is motivated by the observation that CNNs are
sensitive to the objects location when they classify an image [7][8]. Specifically, with a trained CNN model for classification, we can obtain a gradient-based saliency map which can be used to generate candidate object locations in the test image and then objects can be localized by cluttering the salient points with the support of prior knowledge.

The contributions of this paper can be summarized as follows. First, we propose a saliency maps based CBD method. The information in the saliency maps is highly discriminative and our method relies only on image-level labels. Second, we show that CNNs perform well when the objects are not prominent in the images though it is widely believed that the current CNN architectures can only handle the datasets with a single prominent object in the image with limited background clutter, such as CIFAR and ImageNet. We test our proposed method on the Nexar dataset, with promising performance being achieved.

2 Related Work

Recently, CNNs have improved object localization greatly. According to their system frameworks, current CNN-based object localization methods can be divided into three categories: Single Shot Detectors (SSDs), Detect Before Classify (DBC), and Classify Before Detect (CBD).

2.1 Single Shot Detectors
SSDs simultaneously predict the class and the location of an object. They have a set of pre-defined boxes to look for objects instead of having an additional system to produce proposals. Examples in this category include YOLO [5] and SSD [6]. However, they are fully supervised and rely on instance-level labels in training phase.

2.2. Detect-Before-Classify Methods
DBC methods first generate region proposals, and then determine if there is an object existing in each region proposal. Multiple Instance Learning (MIL) is widely used in WSOL, and most MIL-based methods belong to DBC category. Under MIL paradigm, learning usually alternates between two steps. The first step estimates CNN-based object detectors based on a set of training regions, the second stage updates the training set using the learned object detectors. During initialization, region proposals have to be generated, for example, by the most widely-used selective search method [9]. The initialization quality [10] and optimization during iterative training [11] are two major factors for the performance improvement of MIL-based WSOL. Beyond MIL-based methods, end-to-end CNN models are also used for WSOL. Although different training strategies can be used, region proposals are still adopted by several methods [12][13]. In addition, fully supervised methods such as R-CNN series [2][3][4] belong to this category.

2.3 Classify-Before-Detect Methods
CBD methods first perform classification in the whole image, and then use the classification results to assist object detection. Bazzani et al. [14] leverages a classification CNN to localize objects in images using a mask-out strategy. Oquab et al. [15], Zhou et al. [16] and Bency et al. [17] derive object locations from feature maps of classification network directly. Besides, several CBD methods can be used to improve the initial quality of DBC methods. Diba et al. [18] learns to create a class activation map based on object categories to make high quality candidate boxes, and pick the best bounding box by MIL.

This paper proposes a new CBD method for WSOL by leveraging a classification CNN to localize objects using gradient-based saliency maps. The proposed method is efficient because the region proposals generation is not required in the method.

3 A CBD Method for Object Location Based on Gradient-Based Saliency Map

As shown in Fig.1, we present our proposed method. In the classification part, a CNN is trained to determine whether there exist target objects in the image. Then, if there does exist objects, a gradient-based saliency map is generated to outline the candidate locations of objects. Finally, by clustering the salient points, objects can be localized.
Fig. 1. Overview of our CBD method based on gradient-based saliency map.

3.1 Classification
CNNs have significantly improved the performance of image classification, especially in single-label datasets. As it will be discussed in section 3.2.1, as long as the architecture is derivable (most architectures have this property), we can get a gradient-based saliency map. So our method is not strict with the design of CNN architecture. As a result, our method benefits from the development of image classification directly. Nowadays, more and more attempts [19] [20] have been made to make CNN architectures smaller, since small CNN architectures are more feasible to deploy on FPGAs and other hardware with limited memory. Thus, SqueezeNet [19] is selected as the CNN architecture used in this paper. It reduces parameters and computation significantly while maintaining accuracy, with a smart combination of small convolutional kernels and a complex architecture. When an input image is classified to have objects, then the detection part is followed.

3.2 Detection
If target objects exist in the test image, and they belong to class c. Fig.1 has showed a simple illustration for the detection part.

3.2.1 CNN saliencing
There are various methods to generate saliency map based on CNN [7] [21]. In terms of cost, the method proposed by [7] is one of the most efficient. As we all know, in the case of deep ConvNets, the classification score of class c for image I (Sc(I)) is a highly non-linear function of the image I. But in [7], they approximate the Sc(I) using a linear function according to the first-order Taylor expansion:

$$S_c(I) \approx \omega^T I + b$$

where $\omega$ is the derivative of class score $S_c$ with respect to I at the point (input image) $I_0$, just as the equation (2).

$$\omega = \frac{\partial S_c}{\partial I} \bigg|_{I_0}$$

Then, let $I_{ij}$ be the value of the i-th row and j-th column pixel in image $I_0$, it is clear that $| \partial S_c / \partial I_{ij} |$ represents the sensitiveness of $S_c$ with respect to small changes of $I_{ij}$. And $| \partial S_c / \partial I | (I = I_0)$ is the saliency map in [7].

Fig. 2. Saliency map: propagate the gradient to data layer.
As shown in Fig.2, we can get the saliency maps easily in Caffe [22] by propagating the gradient to data layer. As long as the architecture is derivable, saliency maps are available.

Saliency maps indicate which pixels matter for classification. As a result, the locations of the target objects are of high salience in these saliency maps. Besides, saliency maps are learnt from the training
dataset and they are more discriminative than heuristic features. This property will be verified by our experiments.

3.2.2 Salient points clustering and objects localizing

After we have got the gradient-based saliency map, a threshold $t$ is used to binarize the saliency map, and salient pixels whose values equal to 1 are assigned as the candidate locations of objects. However, as the values in saliency maps are discrete, no matter what threshold we choose, it is hard to cluster the salient pixels properly. To cluster the salient pixels, another auxiliary information is necessary. This paper uses the prior knowledge of the target objects. For example, in the traffic light localization case, color can be used as the auxiliary information. In natural images, color presents similarity in areas, it is easy to cluster the same-color pixels if they are connected, and color is useful for traffic light detection. As shown in Fig.3, 8 clusters are clustered after using green-color clustering. Then, we mark the salient points which belong to the same color cluster as a salient cluster, and only 2 salient clusters are retained. Finally, centers of the same-cluster salient pixels can be considered as the locations of target objects. In this process, we can see that the saliency map provides significant guidance to the objects localization.

![Fig. 3. Overview of detection part in our CBD method.](image)

4 Experiments

4.1 Datasets and Metrics

In this paper, we conduct our experiments on the Nexar dataset, the goal of the experiments is to localize traffic lights. Several samples in Nexar are shown in Fig.4. As we can see, street images in this dataset are divided into three classes: none, red and green. Specifically, if there are red and green traffic lights in the scene, it should only identify traffic lights in the driving direction. Note that, unlike most popular datasets for classification such as CIFAR and ImageNet, traffic lights in Nexar occupy only a few pixels in images, which bring great difficulty for their detection using the current object detectors.

We first train a CNN for classification, and test the performance using classification accuracy. Then, the detection results are evaluated using $F$-measure [23], which combines precision $P$ and recall $R$:

$$ F = \frac{2PR}{P + R} \quad (3) $$

Since the ground truths of traffic light locations are not provided in the original Nexar dataset, we select 109 red-class street images and 105 green-class street images from testing set and label them manually, as shown in yellow boxes in Fig.4. All of our experiments are conducted with caffe as the deep learning library. We trained the SqueezeNet model on GPU and tested the model on CPU.
4.2 Implementation details

In our classification experiments, the original training dataset is split into training (17034 images) and testing (1625 images) sets. Stochastic gradient descent (SGD) is adopted to train our network. We run for 100 epoches totally. Note that, in the whole training process, only the associated class labels of training images are utilized.

In the detection experiments, the threshold $t$ for the binarization of the saliency map is set as $0.3 \times \text{max}(M)$ where $M$ is the saliency map value. We use color to cluster the pixels in the original images. In our experiments, color space is simply determined as follows: suppose the red, green and blue values of a pixel be $r$, $g$ and $b$. When $r-g > 20 \& r-b > 20 \& r > 20$, the pixel is considered as red-class, and when $g-r > 20 \& b-r > 20 \& g > 20 \& b > 20$, the pixel is regarded as green-class.

4.3 Quantitative performance

We finally achieve a classification accuracy of 92.75% in the testing set. It illustrates that though the objects are not prominent in the images, CNNs still work.

In the labeled testing dataset, our proposed method extracts 468 traffic lights totally, where 352 are correct results, and 116 are unexpected results. However, 148 traffic lights are missing. $P$ is thus 0.7521, $R$ is 0.7040, and F-measure is 0.7273. Furthermore, if the CNN saliencing is removed, only the color information is used to predict traffic lights locations, F-measure falls to 0.1856. It can be seen that saliency maps are highly discriminative and our CBD method is effective.

In terms of speed, our method takes about 0.16 seconds for classification and 0.23 seconds for generating a saliency map on a single CPU, they can be accelerated by 10-20 times using GPU. The post-processing of our method is straightforward and needs 0.89 seconds. Compared with other methods, many DBC methods [11][12][13] take at least 2 seconds per image because selective search takes about 2 seconds per image; YOLO, known for its speed in SSDs, takes around 6-12 seconds per image on CPU, our proposed method is thus efficient in object localization in CPU implementation. This is mainly due to the simple CNN architecture. In terms of model size, the SqueezeNet model only has 3.0 MB of weights. It is significant smaller than the common models for object localization.

4.4 Qualitative performance

First, we show several saliency maps in Fig.5. All the street images in the figure are classified correctly. As shown in Fig.5, the locations of the traffic lights are of high salience in these saliency maps.
maps. Besides, it is shown that the gradient-based saliency maps have the ability to avoid the interference of tail lights or street lights which is difficult for heuristic features such as color or shape. Thus, the information in saliency maps is more discriminative than that of heuristic features.

**Fig. 5.** Gradient-based saliency maps in Nexar dataset.

Then, as shown in Fig. 6, following section 3.2.2, we can get the final qualitative results. Our proposed method is effective for localizing the traffic lights.

**Fig. 6.** Results of traffic lights localization.

5 Conclusion

In this paper, a new CBD method based on saliency maps is proposed for weakly supervised object localization. We demonstrate that gradient-based saliency maps provide strong resistant capability to interference in object localization. Experiments conducted on the Nexar dataset have verified the effectiveness of our new method.

References

[1] Everingham M, Gool L V, Williams C K I, et al. The Pascal Visual Object Classes (VOC) Challenge[J]. International Journal of Computer Vision, 2010 88(2) 303-338.
[2] Girshick R, Donahue J, Darrell T, et al. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. IEEE Conference on Computer Vision and Pattern Recognition[C]. IEEE Computer Society, 2014. 580-587.
[3] Girshick R. Fast R-CNN. IEEE International Conference on Computer Vision[C]. IEEE Computer Society, 2015. 1440-1448.
[4] Ren S, He K, Girshick R, et al. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks[J]. IEEE Trans Pattern Anal Mach Intell, 2017 39(6) 1137-1149.
[5] Redmon J, Divvala S, Girshick R, et al. You Only Look Once: Unified, Real-Time Object Detection. Computer Vision and Pattern Recognition[C]. IEEE, 2016. 779-788.
[6] Liu W, Anguelov D, Erhan D, et al. SSD: Single Shot MultiBox Detector[M]. Computer Vision – ECCV 2016. Springer International Publishing, 2016. 21-37.
[7] Zeiler M D, Fergus R. Visualizing and Understanding Convolutional Networks. European
Conference on Computer Vision[C]. Springer, Cham, 2014. 818-833.

[8] Simonyan K, Vedaldi A, Zisserman A. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps[J]. Computer Science, 2013.

[9] J. R. Uijlings, K. E. Sande, T. Gevers, et al. Selective Search for Object Recognition[J]. International Journal of Computer Vision, 2013 104 (2) 154-171.

[10] Bilen H, Pedersoli M, Tuytelaars T. Weakly Supervised Object Detection with Posterior Regularization. The British Machine Vision Conference[C]. 2014. 1997-2005.

[11] Cinbis R G, Verbeek J, Schmid C. Weakly Supervised Object Localization with Multi-Fold Multiple Instance Learning[M]. IEEE Computer Society, 2017.

[12] Bilen H, Vedaldi A. Weakly Supervised Deep Detection Networks. Computer Vision and Pattern Recognition[C]. IEEE, 2016. 2846-2854.

[13] Kantorov V, Oquab M, Cho M, et al. ContextLocNet: Context-Aware Deep Network Models for Weakly Supervised Localization. European Conference on Computer Vision[C]. Springer International Publishing, 2016. 350-365.

[14] Bazzani L, Bergamo A, Anguelov D, et al. Self-taught object localization with deep networks. IEEE Winter Conference on Applications of Computer Vision[C]. IEEE, 2016. 1-9.

[15] Oquab M, Bottou L, Laptev I, et al. Is object localization for free? - Weakly-supervised learning with convolutional neural networks. Computer Vision and Pattern Recognition[C]. IEEE, 2015. 685-694.

[16] Zhou B, Khosla A, Lapedriza A, et al. Learning Deep Features for Discriminative Localization. Computer Vision and Pattern Recognition[C]. IEEE, 2016. 2921-2929.

[17] Bency A J, Kwon H, Lee H, et al. Weakly Supervised Localization Using Deep Feature Maps. European Conference on Computer Vision[C]. Springer, Cham, 2016. 714-731.

[18] Diba A, Sharma V, Pazandeh A, et al. Weakly supervised cascaded convolutional networks[J]. arXiv preprint, 2017.

[19] Iandola F N, Han S, Moskewicz M W, et al. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size[J]. arXiv preprint arXiv:1602.07360, 2016.

[20] Howard A G, Zhu M, Chen B, et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications[J]. arXiv preprint arXiv:1704.04861, 2017.

[21] Fu R, Li B, Gao Y, et al. Fully automatic figure-ground segmentation algorithm based on deep convolutional neural network and GrabCut[J]. Iet Image Processing, 2017 10 (12) 937-942.

[22] Badrinarayanan V, Kendall A, Cipolla R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation[J]. IEEE transactions on pattern analysis and machine intelligence, 2017 39 (12) 2481-2495.

[23] Information retrieval: Data structures & algorithms[M]. Englewood Cliffs, New Jersey: prentice Hall, 1992.