Few Shot Object Detection Using Fine-tuning

Jean-Bernard Uwineza
University of California, Riverside
Riverside, CA 92501
buwineza@ee.ucr.edu

Chetan Reddy Mudireddy
University of California, Riverside
Riverside, CA 92501
cmudi001@ucr.edu

Om Shankar Ohdar
University of California, Riverside
Riverside, CA 92501
oohda001@ucr.edu

Sayak Nag
University of California,Riverside
Riverside, CA 92501
snag005@ucr.edu

Vikarn Bhakri
University of California, Riverside
Riverside, CA 92501
vbhak001@ucr.edu

Abstract

Abstract goes here

0.1 Some rules for collaboration

- DO NOT delete text you did not write, unless the person who wrote it asks you. If you have issues with some of the text, discuss with the author first.
- Commit and push to GitHub often. This helps others staying updated.
- Apply other common sense rules of collaboration.

1 Introduction

This text is from the proposal. It will be changed.
Few-shot learning has received significant interest in the past few years, but mainly for the tasks of classification and rarely for object detection. In computer vision, the task of object detection is more challenging since the detector not only has to perform recognition of the different kinds of objects present, it also has to localize them. This is already a challenging task that relies heavily on the availability of massive amounts of labeled training data. Now when a new data-point is obtained belonging to a novel category, adapting the model becomes a very difficult task especially when the new category contains a few samples. Recently, meta learning techniques have been proposed for adapting deep models to novel categories. However, they are not easily extendable to the task of object detection. Take for example the Matching [1] and Prototypical Networks [2], building prototypes of objects is much more difficult than building prototype of the categories. Another approach that is being explored by researchers is to provide ways to fine-tune the detection layers of deep models to adapt to the new categories [3].

In this project we aim to do a comparative study of meta-learning and fine-tuning approaches towards object detection. We aim to experiment on benchmark datasets such as COCO [4] and PASCAL [5] and also extend these approaches towards 3D object detection with the KITTI dataset [6].

Preprint. Under review.

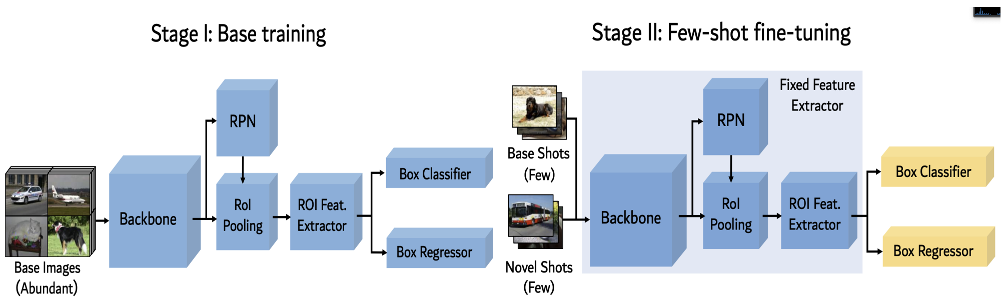
In addition, we plan on examining how many shots are necessary to reach comparable accuracy relative to conventional detection approaches. To this end, we will attempt to develop a metric that assesses the model's knowledge, and requests additional labeled examples if it has not reached a certain accuracy. This will allow us to further compare the performance of the two approaches.

2 Few-shot Finetuning

2.1 Problem Definition

Suppose there is a set of base classes C_b with *sufficiently many* training examples and a set C_n of novel classes with only K examples per class (in this paper $K \leq 10$ always). The few-shot datasets used herein are balanced, meaning that for each novel class there are K annotated examples. When K examples are used in finetuning, we call it a k-shot detection. Denote the set of training images as $\mathcal{X}$ and their corresponding labels as $\mathcal{Y}$. A detection dataset $\mathcal{D} = \{(x, y),\ x \in \mathcal{X},\ y \in \mathcal{Y}\}$, where x is the input image and $y = \{(c_i, \mathbf{1}_i)\}_{i=1}^{N}$ denotes the categories $c \in C_b \cup C_n$ and bounding box coordinates $\mathbf{1}$ of the N instances of the object in the image x.

In [1] and [2], as is usually used in few-shot learning, the authors use N-way K-shot regime for evaluation, where N is the number of training examples, and K is the number of classes in the training set. In this paper, the detector model is evaluated on a test set that includes base and novel classes to optimize the detection accuracy as measured by mean average precision (mAP) of both novel and base classes.



(a) Stage I (b) Stage II

Figure 1: Stage I & II of the finetuning process. Figure courtesy of Wang et al.

2.2 Two-stage Finetuning

In a two-stage process, we adopt the simple finetuning approach of [3]. This approach builds on Faster R-CNN [7], a popular two-stage object detector model. The approach is pictures in Figure 1. Figure 1a shows that the Faster R-CNN model has three main components, namely, the backbone, the Region Proposal Network (RPN), and the two-layer fully-connected network as a Region of Interest (RoI) feature extractor. The backbone used in this paper is the ResNet architecture [8], but it could be any other architecture such as the VGG16. The ResNet was chosen because it is relatively easier to optimize, hence potentially easier to adapt to novel classes. In addition to Faster R-CNN model, henceforth denoted $\mathcal{F}$, the approach also uses a box classifier $\mathcal{C}$ for classifying objects into different classes and a box regressor $\mathcal{R}$ to predict the bounding boxes.

The main intuitive assumption is that $\mathcal{F}$ is able to transfer the feature function $\mathcal{F}(x)$ learned from base classes to novel classes. In other words, $\mathcal{F}$ is class-agnostic. This leads to the main contribution of [3]: a suggestion that when introducing novel classes, it matters little to retrain $\mathcal{F}$. Instead, only the the box classifier $\mathcal{C}$ and regressor $\mathcal{R}$ need to be finetuned on the examples of novel classes since the features of base classes learned with $\mathcal{F}$ will easily transfer to novel classes. In this way, feature representation learning and box prediction learning are separated into two stages, as shown in Figure 1.

Stage I: Base Training. The feature extractor and the box predictor are trained on a dataset containing base classes, C_b. The training loss function [7] is

$$L = L_{\mathcal{F}} + L_{\mathcal{C}} + L_{\mathcal{R}}. \tag{1}$$

The $L_{\mathcal{F}}$ loss is obtained from the output of the RPN network, hence, it helps distinguishing background from foreground. The $L_{\mathcal{C}}$ loss is a cross-entropy loss of the box classifier, and $L_{\mathcal{R}}$ is the robust loss function (smoothed L_1) of the box regressor. More details on L can be found in [7] and [9].

Stage II: Finetuning. In this stage, a small training set containing K examples per class, with both base and novel classes. Using a smaller learning rate[1] and the same loss function as in (1), the box classifier and regressor are initialized with random weights for the novel classes and are finetuned while the feature extractor $\mathcal{F}$ is fixed and unmodified.

2.3 Euclidean Distance-based Similarity

The weight matrix $W \in \mathbb{R}^{d\times c}$ of the box classifier $\mathcal{C}$ can be expressed as $W = [w_1, w_2, \ldots, w_c]$ where the row-vector $w_c \in \mathbb{R}^d$ is the weight vector of each class. Recent works (such as [1, 10–12]) have proposed box classifiers based on cosine similarity. The output of the classifier is a scaled similarity score matrix of the input features and the weight vectors of the classes. Specifically

$$s_{ij} = \frac{\alpha\mathcal{F}(x)_i^\top w_j}{\|\mathcal{F}(x)_i\|\|w_j\|}, \tag{2}$$

where α is the scaling factor[2]. Cosine similarity is attractive because it offers feature normalization which reduces the variance between classes and improves detection accuracy for novel classes [3].

We now propose a different similarity score, in which the Euclidean distance between the input features and class weight vectors are passed through the Softmax function. Specifically

$$s_{ij} = \frac{e^{-\|\mathcal{F}(x)_i - w_j\|_2^2}}{\sum_j e^{-\|\mathcal{F}(x)_i - w_j\|_2^2}}. \tag{3}$$

This similarity score provides a stronger kind of feature normalization which should help the model adapt to novel classes when introducing novel classes.

3 Other Sections

4 Experimental Results

5 Conclusion

References

[1] O. Vinyals, C. Blundell, T. P. Lillicrap, K. Kavukcuoglu, and D. Wierstra, "Matching networks for one shot learning," *CoRR*, vol. abs/1606.04080, 2016. [Online]. Available: http://arxiv.org/abs/1606.04080

[2] J. Snell, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," in *Advances in neural information processing systems*, 2017, pp. 4077–4087.

[3] X. Wang, T. E. Huang, T. Darrell, J. E. Gonzalez, and F. Yu, "Frustratingly Simple Few-Shot Object Detection," *arXiv preprint arXiv:2003.06957*, 2020. [Online]. Available: http://arxiv.org/abs/2003.06957

[4] T. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: common objects in context," *CoRR*, vol. abs/1405.0312, 2014. [Online]. Available: http://arxiv.org/abs/1405.0312

[1] The finetuning learning rate is reduced by at least 20 times compared to base training learning rate. In our experiments, it was reduced from $2 \cdot 10^{-2}$ to $1 \cdot 10^{-3}$.

[2] In [3], a fixed value of $\alpha = 20$ was used.

[5] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, “The Pascal Visual Object Classes (VOC) Challenge,” *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, Jun. 2010.

[6] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, “Vision meets Robotics: The KITTI Dataset,” *International Journal of Robotics Research (IJRR)*, 2013.

[7] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” in *Advances in neural information processing systems*, 2015, pp. 91–99.

[8] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[9] R. Girshick, “Fast R-CNN,” in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

[10] W.-Y. Chen, Y.-C. Liu, Z. Kira, Y.-C. F. Wang, and J.-B. Huang, “A closer look at few-shot classification,” *arXiv preprint arXiv:1904.04232*, 2019.

[11] H. Qi, M. Brown, and D. G. Lowe, “Low-shot learning with imprinted weights,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 5822–5830.

[12] S. Gidaris and N. Komodakis, “Dynamic few-shot visual learning without forgetting,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4367–4375.