Deep Learning Challenge for Master's Degree at Unicamp - Rethinking the Route Towards Weakly Supervised Object Localization

First A. Bruno César de Oliveira Souza*
Mechatronics Student, Ribeirão Preto, SP

These paper aims to explain the development of the programmation associated to the article "Pseudo supervised object localization" (PSOL). PSOL is a method that tries to be an alternative to the Weakly supervised object location (WSOL) in object detection domain. The work presented on this paper is a programmation recognition challenge and, therefore, it is not intented to acquire highly accuracy in their prediction model or be a deep learning technique in the real world for researchers or students.

I. Introduction

Deep convolutional learning needs a large amount of data to perform object detection taks. To be a more flexible method of classification, detection and segmentation, Weakly supervised object location is proposed [1]. WSOL is a modern technique that tries to perform object detection with only image-level labels. By other hands, PSOL aims to be a object detection method in a "class-agnostic" way, which means that is not related to a classification labels. For this goal, PSOL divides its task into two independent subtasks: localization objects without classification and the classification task, separately.

The CUB-200-2011 dataset and its annotation of bounding box groundtruth are used during the development of the PSOL method. Due to the time and computational resource limitation, only two class are selected amoung all 200 class. The code developed in python is inspired by the official code from the original paper and can be find on its github page : `https://github.com/tzzcl/PSOL`.

The classification performances on CUB-200 dataset presented around 95% accuracy due to transfer learning technique. However the bounding box prediction has an insignificant performance of 16.67%. The low performance came from the limit amount of data for training the model.

The code was developped in Python language using the Pytorch, a deep learning framework. In order to achieve the goal, the code is divided in three main blocks as explained below:

1) Generate a class-agnostic pseudo bounding boxes with a class-agnostic method.
2) Fine-tune a pretrained model for a regression bounding box prediction.
3) Fine-tune a pretrained model for a classifition task.

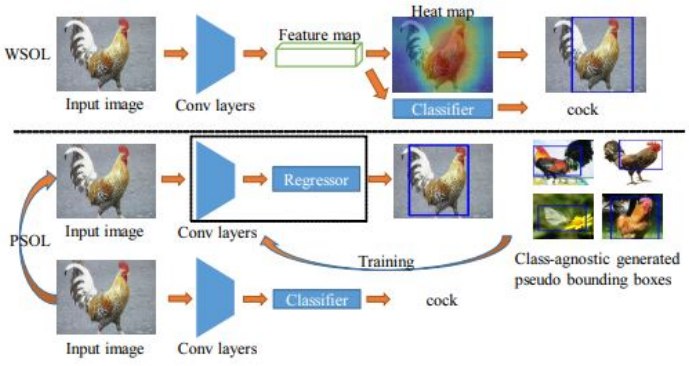

Fig. 1 WSOL technique (top) and PSOL technique (bottom).

*Mechatronics Student, brunocosouza@gmail.com.

II. PSOL x WSOL methods

[2] proposes a paradigm shift for the weakly supervised object localization.According to [2] PSOL achieved state-of-the-art performances in both ImageNet1k and CUB-200 dataset. Obtaning the result from the two independent sub-task, PSOL combining the results of these two independent sub-tasks and achieve a large edge over WSOL models.[2] presents that with classification results of the recent EfficientNet model, they achieved 58.00% Top-1 localization accuracy on ImageNet1k, which significantly outperforms previous methods.

WSOL localizes objects with only image-level labels. For this task, WSOL assumes that there is only one main object in the whole image. However, this methods has a strong class-label dependency for its goal. These limitations become WSOL a non-ideal method to real-world problems.

In order to push those limitation, PSOL is proposed. The key idea of PSOL is divide WSOL into two tasks: class-agnostic object localization and object classification. The first sub-task use a class-agnostic method of Deep Description Transforming (DDT) to generate pseudo bounding boxes allowing to perform object localization. Using the pseudo bounding boxes, a class-agnostic model tries to predict localization of the object. The second sub-task needs a CNN (or its derivatives) to classify the object represented in the image. Finally the result from the both models are combined to form the final prediction.

Table 1 PSOL Algorithm

Algorithm 1:	Pseudo Supervised Object Localization
Input:	Training images I_{tr} with class label L_{tr}
Output:	Predicted bouding boxes b_{te} and class labels L_{te} on testing images I_{te}
1:	Generate pseudo bounding boxes b_{tr} on I_{tr}
2:	Train a localization CNN F_{loc} on I_{tr} with $\widetilde{b_{tr}}$
3:	Train a classification CNN F_{cls} on I_{tr} with L_{tr}
4:	Use F_{loc} to predict b_{te} on I_{te}
5:	Use F_{cls} to predict L_{te} on I_{te}
Return:	b_{te}, L_{te}

III. Methodology and Results

A. Dataset

We used the CUB-200 database to perform the PSOL method. Due to the computation resource limitation, only the first two class were applied to train our model. The training file is composed of 48 images for each class and the validation file gathers 12 images each class as well. The bounding boxes for the training data are the ones we generated with the DDT methods. However, for comparing, the ground-truth bounding boxes used in the validation dataset come from the annotation provided by CUB-200.

B. Implementation Details

The first step of PSOL method is to generate the class-agnostic pseudo bounding boxes. In order to achieve this goal, the DDT techniques are used. [3] shows that DDT presents a good performance without a high computational resource. This method achieved a good result. In several images, the bounding boxes wrapped the whole object presented in the images. Only a few samples presented an acceptable bounding box, which means they limited not the whole object, but it could find the main features. Figure 2 shows the result.

The second step of the process is achieved by training our regression model for localization object. We downloaded a ResNet pretrained model and added a fully connected layer with ReLU layer for regression and the output came from a sigmoid activated. We used a 12 batch-sized, 0.001 of learning rate, momentum of 0.9 in 10 epochs for training. As the prediction is yielded by a regressor the metric used to backpropagation is the mean squared error loss (l2 loss). Despite our best accuracy in validation is around 16%, the Figure 3,4 below show that our model seems to keep learning because the accuracy curve is growing (even though it is a noisy curve) while the loss curve continues to decline.

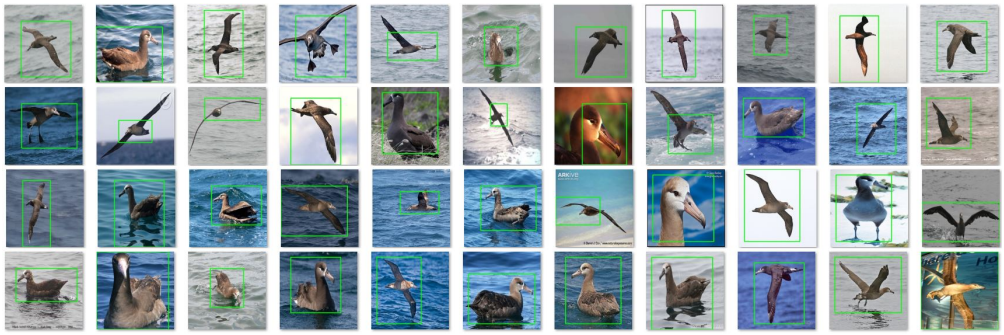

Fig. 2 Bounding boxes generated by a class-agnostic method. In this case is used the DDT method.

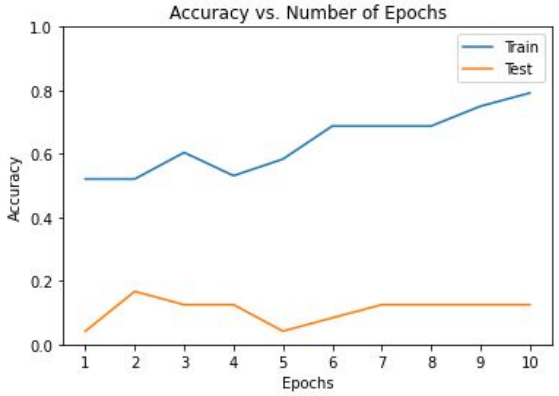

Fig. 3 Accuracy curve of training and test for bounding boxes regression predictor

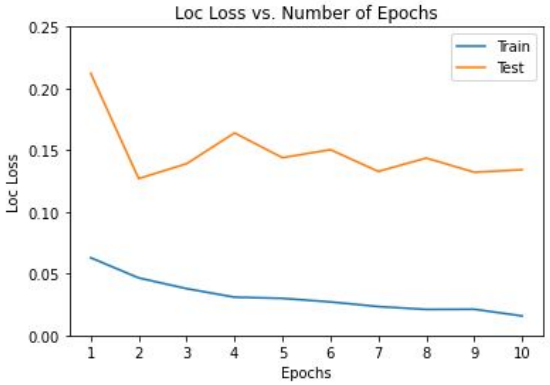

Fig. 4 Loss curve of training and test for bounding boxes regression predictor

The classification step has a place in the third process of the PSOL method. For a better approach, we applied the transfer learning strategy. We choose a classification ResNet architecture model available in the pythorchvision package. The Resnet18 model was downloaded and the fully connected layer was reshaped to make sure the output is the number of class we have, two in this case. For the model criterion loss we setup the cross-entropy approach. After the training process, the transfer learning strategy presents a good accucary in our classes. This good approach is due to the pre-trained model that already has a good feature extraction.

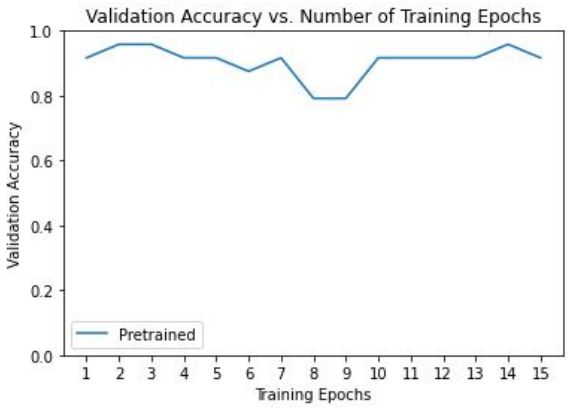

Fig. 5 Validation Accuracy vs. Number of Training Epochs

C. Inference

The inference is the last part of our code. Both models receive an image as input and the outputs generated are the bounding box and the classification relative to the image. As we can see in Figure 6, our regressor model has a little difficult to predict the best place for the bounding box, even though it has found a small fragment of our object. On the other hand, our classification model has good accuracy as we can confirm in the images.

Fig. 6 Inference

References

[1] Miaojing Shi, V. F., Holger Caesar, "Weakly Supervised Object Localization Using Things and Stuff Transfer," *Computer Vision Foundation*, 2017.

[2] Chen-Lin Zhang, J. W., Yun-Hao Cao, "Rethinking the Route Towards Weakly Supervised Object Localization," *National Key Laboratory for Novel Software Technology*, 2020.

[3] Xiu-ShenWei, J. C. S. Z.-H. Z., Chen-Lin Zhang, "Unsupervised object discovery and co-localization by deep descriptor transformation," *Pattern Recognition*, 2019.