BAOD: Budget-aware object detection

Alejandro Pardo

Department of Biomedical Engineering

University of the Andes

Supervisor

Pablo Arbelaez

In partial fulfillment of the requirements for the degree of

Master of Science in Biomedical Engineering

March 28, 2019

BAOD: Budget-Aware Object Detection

Alejandro Pardo and Pablo Arbeláez

Universidad de los Andes

Abstract—We study the problem of object detection from a novel perspective in which annotation budget constraints are taken into consideration. When provided with a fixed budget, we propose a strategy for building a diverse and informative dataset that can be used to optimally train a robust detector. We develop both optimization and learning-based methods to sample and annotate images, and we adopt a hybrid supervised learning framework to train the object detector from both image- and instance-level annotations. We conduct a comprehensive empirical study showing that a handcrafted optimization method outperforms other selection techniques including random sampling, active learning, and reinforcement learning. By combining an optimal image selection/annotation scheme with hybrid supervised learning, our proposed budget-aware approach can achieve the performance of a strongly supervised detector on PASCAL-VOC 2007 while saving 12.8% of its original annotation budget. Furthermore, when 100% of the budget is used, our approach surpasses this performance by 2.0% mAP.

Index Terms—Object Detection, Convolutional Neural Networks, Active Learning, Annotations and Datasets.

I. INTRODUCTION

OBJECT detection in images is a fundamental computer vision problem with applications in many tasks including face/pedestrian detection [1], [2], [3], [4], [5], [6] counting[7], [8], and visual search [9], [10]. The classical approach for the detection problem consists of localize all the instances of a specific object of interest in the image. In the same way, object detection can be seen as a binary classification problem with very unbalanced classes. The localization can be indicated either with a point in the center of the object or with a bounding box around it. Due to the unbalanced nature of the problem it is evaluated with Average Precision (AP) obtained as the area under the curve of the Precision-Recall curve (PR Curve). If the dataset contains more than two classes the AP is evaluated for each class and the performance is reported as the mean between all the APs, this metric is called Mean Average Precision (mAP). Precision is defined as the true positive ratio between all object candidates, while recall is defined as the True positive ratio between all the objects in the image. To obtain the PR Curve, we need to get object candidates with a confidence score. Then, a threshold of the confidence measure is applied to obtain a data point in the PR curve.

Building an object detector encompasses three main dimensions: (1) **the image dataset** to be annotated mainly for training but also for testing the detector. A larger dataset allows for a more accurate detector, but the number of training images is limited by the *annotation budget*. Even when a large amount of labels is freely available, budget is also limited by computational complexity [11].; (2) **the annotation scheme** used to label the training images. One could annotate either

Fig. 1: Budget-Aware performance of detectors with different levels of supervision. X axes show performance of the model in terms of mAP, and Y axes shows different Budget levels in terms of Budget units N. The models are trained on PASCAL VOC 2007 trainval (VOC07), PASCAL VOC 2012 trainval (VOC12) as specified in the legend. Our proposed Budget aware object detection (BAOD) (green ∇ - curve) has a higher mAP than FSOD (yellow $+$ - curve) and WSOD (orange $-$ - curve) methods at most budgets. Given a larger unlabeled image pool (Blue $\diamond$ - curve, VOC07+VOC12), our BAOD can reach a higher mAP using the same budget needed to annotate VOC07 with instance-level labels. Since the dataset is finite, WSOD cannot increase its performance with more budget.

image-level labels (the categories of the objects are known but their locations are unknown) or instance-level labels (both categories and locations are known) [4], [12], [13], [14], this level of labels is also now as levels of supervision; (3) **the detection model**. Most work on object detection fixes the first two dimensions and only explores the third. In fact, they tend to focus on optimizing the detection model based on one or more training datasets that are usually annotated with the same kind of annotations. We fix the detection model and seeks an optimized solution in the first two dimensions of the problem.

A large group of object detectors falls under the umbrella of Fully-Supervised Object Detection (FSOD) methods [15], [16], [17], [18], [19], [20], [21], [22]. It has been shown in recent years that these techniques can reach high detection performance, especially with the introduction of Convolutional Neural Networks (CNNs) [23], [24]. However, FSOD requires large datasets with strong annotations (*i.e.* instance-level annotations with object category and location) like [25], [26], [12]. This requirement makes FSOD methods expensive and time

consuming to scale. For instance, the official protocol used to annotate ILSVRC [25] requires about 30 seconds per object box [27].

In contrast, Weakly-Supervised Object Detection (WSOD) [28], [29], [30], [31], [32], [33], [34], [35], [36], [37], [38] aims at building object detectors from cheaper but less informative image-level annotations (*i.e.* only object categories are given). These weak annotations limit the detection performance of WSOD methods when compared to their FSOD counterparts trained on the same image dataset but with instance-level annotations. In fact, reaching FSOD performance is the gold standard for current WSOD methods. However, this comparison is inherently unfair, since WSOD training receives less information overall than FSOD, thus, registering a clear advantage for the latter.

In this paper, we argue that a fairer means of comparison is needed to put detectors on the same footing when different annotation schemes are used. Ideally, detectors should only be compared when they are trained using image datasets that offer the same amount of information (not necessarily the same number of images). Since this notion is difficult to define quantitatively, we take the **training budget** of a detector as a unifying surrogate measure. Here, we define budget as the effort, or cost, to annotate a dataset, thus, combining the first two dimensions of the detection problem: dataset scale and annotation scheme. In fact, with a fixed budget and a set of unlabelled images, the number of images we can label depends highly on the annotation cost. Such cost varies significantly between image- and instance-level annotations. Typically, annotating a bounding box around an object in an image is more expensive than simply annotating its category [4], [25]. Therefore, an FSOD method with the same budget as a WSOD method contains fewer images in its training set.

Consequently, a fairer comparison between object detection methods should focus on their performance when using the same budget. Figure 1 compares a FSOD [20] and a WSOD method [39] at various budget levels, where the lowest budget corresponds to labelling the entire PASCAL VOC07 dataset with image-level annotations and the highest budget corresponds to the cost of labeling it with instance-level annotations. We observe that the WSOD method outperforms the FSOD one at low budgets. This result makes intuitive sense, since a low budget corresponds to many weakly annotated images and a much smaller number of strongly annotated ones, favoring WSOD. When the budget increases, the FSOD performance exceeds the one of the WSOD method.

We explore strategies to build better object detector models when constrained with a training budget, a novel problem we coin *budget-aware object detection*. As such, we focus on the first two dimensions of the detection problem: choosing the best images for training and how to annotate them. To explore these two dimensions, we develop an active learning pipeline that sequentially chooses both the image and type of annotation, based on a selection function. Additionally, we propose a novel hybrid training procedure for object detectors that can use both types of annotations: strong (instance-level) and weak (image-level). The combination of the active learning pipeline and the hybrid training is our Budget-Aware

Object Detection (**BAOD**). Figure 1 shows that this method outperforms FSOD and WSOD when the budget is larger than 20% of the budget needed to annotate VOC2007 trainval at the instance-level. The yellow curve also shows that we can reach a higher mAP if we use the same budget of VOC07 to annotation both datasets VOC07 and VOC12.

a) Contributions.: **(1)** We present a new evaluation criterion for object detection algorithms. This criterion takes into account not only detection performance but also budget. **(2)** We propose a hybrid supervised learning strategy that combines category and localization information to train a robust detection model. This framework inherits the advantages of both WSOD and FSOD methods while avoiding their drawbacks. **(3)** We use active learning to study the selection process of both training images and their annotation scheme. We compare various solutions to the selection problem, including optimization and learning-based methods. **(4)** By combining intelligent image and annotation scheme selection with hybrid supervised learning, we train a model that can outperform the state-of-the-art in two complementary ways. We improve performance on VOC07 by 2% mAP combining both VOC07 and VOC12 train-val sets but using the same budget as the one used in VOC07. Also, we can achieve the current state-of-the-art performance on VOC07 [40], while saving 12.8% of its budget.

II. RELATED WORK

A. Fully Supervised Object Detection (FSOD)

With the development of deep learning, many CNN based methods have been proposed to solve the FSOD problem, such as Fast R-CNN [18], Faster R-CNN [20], SSD [41], YOLO2 [16], and their variants [19], [21], [22].

Faster R-CNN [20] is a typical proposal based detection CNN, which balances both detection performance and computational efficiency. It is based on two main states, one for class-agnostic object proposals and the other one for classification and bounding box refinement. The proposal network is called *Region Proposal Network (RPN)*, it receives as input a feature map from several convolutional layers and outputs a fixed number of proposals each with its associated *objectness* score, which indicates the probability of each box to contain an object of any class.

To generate this proposals, RPN performs a sliding window procedure across the feature map. For every position it considers a certain number of boxes with different scales and ratios, called *anchors*. Then, each anchor is classified as background or foreground and its position is refined. The foreground anchors become the object proposals to consider in the next stage. The next stage takes the object proposals and classifies and refines its locations to give the final predictions. Due to the different size of each object proposal, Girshick *et al.* [18] proposed a Region Of Interest Pooling (RoI Pooling), in order to convert each object proposal into a fixed size feature map to allow the use of fully connected layers. RoI pooling splits the input into a fixed number of regions ($k \times k$) and applies max-pooling on every region. After RoI pooling, each proposal is classified and refined using two fully convolutional layers. A brief overview of faster-R-CNN is showed in 2.

Fig. 2: Overview of faster-RCNN method. Where we can see a first stage called Region Proposal Network to generate candidates, and a second stage to use this proposals by applying RoI Pooling, to classify and generate the final predictions. Image taken from original paper [18].

This method has become the *de facto* framework for fully-supervised object detection. Although FSOD methods have achieved impressive results in object detection, training them is expensive due to the collection of large-scale instance-level bounding-box annotations is expensive and time-consuming. In this paper, we constrain the budget and focus on how to reduce required annotations without compromising the object detection performance. Su *et al.*[27] reported that it takes 26 seconds for drawing one bounding-box without quality control, and 42 seconds with quality control. There has been recent progress in developing tools to reduce annotation time (most recently to 7 seconds) [14], [4], [13]. Our work investigates a complementary aspect of annotation to emphasize that some images do not need to be strongly annotated to achieve a good performance. Our discussion of budget-aware object detection only considers the labour cost for image annotation, but training detection models also consume computational resources. Fortunately, the hardware development and resources re-organization are getting cheaper day by day, while human resources are getting more expensive.

B. Weakly Supervised Object Detection (WSOD)

If there are no instance-level labels available in training, we can switch to training a weakly supervised object detector. Most classical approaches cast WSOD as a Multiple Instance Learning (MIL) problem [42], [34], [35], [36], [28], [38]. Bilen

et al.[31] firstly utilized a deep convolutional neural network to solve this problem. They present a weakly supervised deep detection network (WSDDN) to select positive samples by multiplying the score of recognition and detection. Tang *et al.*[43] improved upon WSDDN by designing an online instance classifier refinement algorithm to alleviate the local-optimum problem that plagues WSDDN. Zhang *et al.*[39] proposed a simple yet effective post processing step to excavate pseudo ground truth bounding boxes used for FSOD training.

We use the weak supervision to improve pseudo object labels. These labels are generated from a previously trained detector, and are cleaned from a pseudo label mining process. The image labels help to remove false predictions that correspond to a wrong object category.

C. Hybrid Supervised Learning

In our work, we study hybrid supervised learning for object detection. This type of learning exploits multiple types of supervision during training. Semi-supervised learning is a special case of this family, since it learns a model from a set of labeled and unlabeled data. Several works [44], [45], [46], [47], [48] try to solve generic semi-supervised learning from *teacher-student learning*. They firstly train a *teacher* model from the fully annotated subset, then the predictions obtained from the teacher model are used to regress a second model that is called the student model. More generally, Hu *et al.*[49] proposed a novel weight transfer function to solve the instance segmentation problem from a hybrid dataset of instance- and pixel- level annotations. Chéron *et al.*[50] also used several kinds of annotations to train an activity recognition model, and they revealed that fully annotating every training sample is not necessary to achieve remarkable localization results.

Our hybrid dataset consists of image- and instance-level annotations. Inspired by Rosenberg’s work [48], our teacher detector is learned from instance-level annotations, while the student detector are learning from ground-truth or processed pseudo object labels.

D. Active Learning

In general, active learning is a sequential decision making process that iteratively estimates the most useful examples to annotate by an oracle and to add to the training set. It aims at training more accurate models with the minimum data required. This field has been widely studied in the context of image and video classification [51], [52], [53], [54], [55], [56], object and action localization [57], [13], [58], [14], [59], human pose estimation [60] and visual question answering [61]. As specified in previous work, active learning aims at maximizing performance while minimizing the human cost in labeling the training samples [51], [60], [57].

The budget-aware object detection problem can also be formulated as a sequential decision making process. However, we have two active processes: (1) choose the next batch of training images and (2) decide which level of annotation. In doing so, we focus on the annotation sequence that can provide the most useful information to the detector. To the best of our knowledge, we are the first to use active learning and hybrid training to study object detection.

Fig. 3: Overview of the whole active learning pipeline to construct an hybrid dataset. Image pools are illustrated with circular shapes, while methods for training and choosing images are illustrated with rectangles. Solid arrows denote information flux, while dashed arrows denote actions.

III. APPROACH

In this section, we explore several ways to use active learning to create an object detection dataset, which includes both image-level and instance-level annotations. Then, we introduce a hybrid supervised learning procedure to take advantage of this mixed labelled training set. A complete overview of the whole active learning procedure is shown in Figure 3.

A. Hybrid Object Detection Dataset using Active Learning

Our hybrid supervised dataset should consider which images to include in training and how to label them. To tackle this problem, we propose an active hybrid learning framework. The framework acts as an active learning agent to simulate the annotation process. At every active step t , we have a budget constraint d to spend on annotation. Each image in this set belongs to one of three pools: unlabeled U_t , weakly labeled W_t , or strongly labeled S_t . With the initial S_t we train an initial model. We use the model to estimate a selection function to take an action on U_t or W_t pools. As shown in Figure 3, we have three possible actions with their associated cost: (1) annotate strongly $\mathbf{x}_1$ (i.e. send the image from U_t to S_t), (2) annotate weakly $\mathbf{x}_2$ (i.e. send the image from U_t to W_t), or (3) strongly annotate a weakly annotated image $\mathbf{x}_3$ (i.e. send the image from W_t to S_t). Then, these images are used to train a new model. We proceed iteratively until we either run out of images in $P_t = U_t \cup W_t$ or run out of budget d .

We study three active learning **sampling functions** (Random, Uncertainty and Learning Active Learning) within five action **selection methods**: Random Sampling (RS), Uncertainty Sampling (US), Optimization based on US, Optimization based on Learning Active Learning (LAL), and Reinforcement Learning (RL).

For RS, we randomly choose an *active batch* of images to include into S_t and W_t . For the US case, we sort each image in descending order based on an uncertainty score denoted as s_k , and pick the top images to include into S_t and W_t . We follow the regular annotation priorities in the RS and US set-ups, thus in the first steps we weakly label U_t , then within the remaining steps we strongly label W_t . There are several ways to evaluate

the s_k of an unlabeled image [62], [4]. We compute an entropy measure for each bounding box of an image using the classification score predicted by the current detection model, and then average them to compute the uncertainty score of the image. Collecting M positive predictions from each image I_k , each prediction has a probability score vector $\mathbf{p}_i \in [0, 1]^c$ for the c positive categories. We compute the uncertainty score as in Eq. 1:

$$s_k = \frac{1}{M} \sum_{i=1}^M \sum_{p \in \mathbf{p}_i} -p \log(p) \quad (1)$$

The next three subsections explain the remaining active selection methods: Optimization based on US, Optimization based on LAL, and RL.

1) *Optimization Based Selection Policy using US*: At t -th active step, let $P_t = \{I_k\}_{k=1}^N$ be the N images that can be further annotated, and $\mathbf{s} \in \mathbb{R}^N$ be the corresponding uncertainty scores. We have three possible actions vectors $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3 \in \{[0, 1]\}^N$. If the k -th element of $\mathbf{x}_i$ is 1, we annotate the image I_k by option i . Assume the next active batch of annotations have a linear impact on the model performance increment δ_t .

$$\delta_t = f_1(P_t)^\top \mathbf{x}_1 + f_2(P_t)^\top \mathbf{x}_2 + f_3(P_t)^\top \mathbf{x}_3 \quad (2)$$

To quantitatively show the contribution of new images, we also assume that the uncertainty score is a complete statistic of a unlabeled or weakly labeled set of images. Then we can approximate the three functions f_1, f_2, f_3 by proportional functions with input $\mathbf{s}$. Most studies [63], [57] experimentally show that appending images with a higher uncertainty score can further improve the detector, but such images can also increase the difficulty to give a true positive prediction. Intuitively, we give positive coefficient for actions $\mathbf{x}_1, \mathbf{x}_3$ which end up with strongly annotated images, and a negative one for weakly annotation $\mathbf{x}_2$. In function f_2 , we shift the scores distribution to be balanced around the origin by subtracting its median value μ .

$$\delta_t \approx \mathbf{s}^\top (\mathbf{x}_1 + \mathbf{x}_3) + (\mu - \mathbf{s})^\top \mathbf{x}_2 \quad (3)$$

Consider the constraints, we define ψ to distinguish images' annotation state. $\psi_k = 1$ only if the k -th image has already weakly annotated. For each solution, we have only one action for each image, and $\mathbf{x}_3$ (strongly annotate from weakly labeled data) can be positive only with a positive ψ_k . In the budget constraints, if we use a budget d in each active step, the cost, $\mathbf{1}^\top (a\mathbf{x}_1 + b\mathbf{x}_2 + c\mathbf{x}_3)$, in such active step should be in the range of $[d - a, d]$. Here a, b, c are the cost to each labeling method, and d is the given budget for each active step.

In conclusion, we are going to solve the following constrained integer optimization problem to maximize the expected performance increment. The first two budget constraints force running out of the budget. The last three constraints make our solution consistent to actual operations.

$$\begin{aligned}
& \max_{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3 \in \{0,1\}^N} \quad \mathbf{s}^\top (\mathbf{x}_1 + \mathbf{x}_3) + (\mu - \mathbf{s})^\top \mathbf{x}_2 \\
& \text{s.t.} \quad \mathbf{x}_3 \leq \psi \\
& \quad \mathbf{x}_1 + \mathbf{x}_2 \leq \mathbf{1} - \psi \quad (4) \\
& \quad \mathbf{1}^\top (a\mathbf{x}_1 + b\mathbf{x}_2 + c\mathbf{x}_3) \leq d \\
& \quad \mathbf{1}^\top (a\mathbf{x}_1 + b\mathbf{x}_2 + c\mathbf{x}_3) \geq d - a
\end{aligned}$$

2) *Optimization Based Selection Policy using LAL*: The Uncertainty score $\mathbf{s}$ is not the only selection function that have been studied in active learning domain. We also follow the Learning Active Learning (LAL) method proposed by Konyushkova *et al.*[53], but take the increment of the absolute mAP Δ as a learned target. Given an image $I_k \in P_t$ and the current detection model, we build a feature vector $\mathbf{v} = [O_t, s_k]$ that concatenates both the model state and the image representation. We keep using $\mathbf{s}$ as the image representation, and the model state O_t is the average precision curves under 5 different Intersection over Union (IoU) thresholds. Starting from an initial model trained with an initial set of images, we aggregate a new image to get a new model and check the absolute increment Δ in mean average precision. This Monte-Carlo process uses the same procedure as described in [53]. After that, a Support Vector Regression is used to learn two functions h_s and h_w that map from the feature vector $\mathbf{v}$ to Δ for weakly and strongly annotations, respectively.

Once we have the two SVRs, the images in the P_t are evaluated and produce two new vectors denoted as $\mathbf{h}_s$ and $\mathbf{h}_w$. The uncertainty score in Eq.4 is replaced by the new vector and then we have another objective function Δ_t to maximize.

$$\Delta_t \approx \mathbf{h}_w^\top \mathbf{x}_2 + \mathbf{h}_s^\top (\mathbf{x}_1 + \mathbf{x}_3) \quad (5)$$

For every active step t , we solve Equation 5, following the same constraints defined in Section 4. In this case, we also follows the procedure illustrated in Figure 3.

3) *Reinforcement Learning*: The problem of finding the next active batch of training examples can be naturally formulated as a reinforcement learning problem. This approach allows us to learn a strategy directly from trial-and-error experience and to avoid the specific mathematical model of the previous sections. The reinforcement learning methods do not need any prior knowledge about the environment, but they can still construct an optimal strategy. Thus, it is easily extensible to stochastic environments with variable set-ups such as strongly-weakly annotation costs ratio and a total amount of budget.

Suppose that we already have a training set, and we regressed a detector with performance m_0 . In the next active step, we evaluate the uncertainties of a subset of unlabeled images. The active learning agent interacts with the environment in the form of the **difficulty preference** and **annotation type**. We still use O_t as defined in section III-A2 as the observed state, which considers both the classification and localization performance. The agent returns two actions from current state: (1) asking for low/medium/high uncertainty images and (2) giving a strongly/weakly annotation. In conclusion, the RL agent gives a choice a from *three by two* possible actions matrix.

Fig. 4: Illustration of the Hybrid Learning framework. Given an image collection with hybrid labels, we firstly use a warm-up detection model to generate pseudo instance labels (blue dash rectangles in the top image.) After cleaning noise, the second detection model learns from both the ground truth and pseudo instance labels.

In the RL training process, we take the detection model increment performance Δ_t as a reward. If there is still enough budget, the agent chooses the next active batch from the new observed state O_{t+1} . If we run out of budget after T active steps, the total reward is $R = \sum_{t=1}^T \Delta_t = m_T - m_0$. Here m_0 is the performance of a randomized warm-up model with a fixed seed. By maximizing the return R , the agent learns a policy π that maximises the final model performance m_T .

To train the agent, we learn a policy with Q-learning[64] which learns to approximate Q-function $Q^\pi(a, O_t)$ with a three-layer neural network. Given an action a and the current state O_t , it returns the expected value from policy π .

B. Hybrid Supervision for Object Detection

In the hybrid supervised learning framework, we firstly train an initial detector with the strongly labeled image set S_t . Then, we use the detector as a teacher model to generate pseudo instance labels associated with their confidence scores. After that, we employ pseudo label mining to reduce the instance label amount and increase their precision. Finally, both pseudo labeled images and strongly labeled images are used to train a student model. Figure 4 gives a brief overview of our hybrid learning pipeline.

Pseudo Label Mining Pseudo label mining aims at cleaning redundant or erroneous pseudo labels from the teacher model's predictions. We use the minimal knowledge to post-process the predictions and focus our study on the comparison of active learning methods and the benefit of hybrid training. Given a weakly labeled image I_k with the annotation $A \in \{0, 1\}^{N \times K}$, the teacher model returns M positive predictions for the localization $P \in [0, 1]^{N \times 4}$, classification $\omega \in \{1, 2, \dots, c\}^N$ and the positive class confidence vector $\mathbf{p}_m \in [0, 1]^N$. We use weak supervision to remove false alarms, and keep the relatively high confidence predictions when there is a strong overlap. Eventually, the pseudo label mining returns an sparse M -dimensional binary vector $\mathbf{y}$ as Eq.6.

$$\begin{aligned}
& \min_{\mathbf{y} \in \{0,1\}^M} \quad -\mathbf{y}^\top \mathbf{p}_m \\
& \text{s.t.} \quad \mathbf{y}^\top A(\mathbf{1} - \mathbf{w}) = 0 \\
& \quad \forall_{y_i, y_j=1} \text{IoU}(P_i, P_j) \leq \alpha \\
& \quad |\mathbf{y}|_0 \in [1, \beta] \quad (6)
\end{aligned}$$

This objective function maximizes the confidence score for all pseudo labels. In mean time, we check the image label consistency from the first constrain and remove highly overlapped predictions from the second constrain. Moreover, the last constraint force the pseudo labels number between 1 and β to have a precise result. The optimum value y is a binary sampling vector in which the i -th positive value means taking i -th predictions as a pseudo label. We set $\alpha = 0.3, \beta = 3$ to make the pseudo labels more precise.

IV. EXPERIMENTS

A. Experimental Setup

Datasets. As usual in WSOD, we use the PASCAL VOC dataset [26] to perform most of the experiments. Given the active learning pipeline of our method, we emulate the oracle using PASCAL ground-truth. The trainval set of PASCAL VOC 2007 (VOC07) and 2012 (VOC12) have 5011 images and 11540 images, respectively. These images are designed to detect objects in 20 categories. Our models are either trained on VOC07 trainval set or on both training sets of VOC07 and VOC12. These models are evaluated on the VOC07 test dataset. We assume that the cost of a strong label (strong cost) is: *i.e.* 34.5 seconds per instance [25] for every experiment. The annotation time for each weakly label (weak cost) is 1.6 second per instance [25] on average. According to a statistical report [26], each image has 1.37 objects. We take the total strong cost D of VOC07 trainval as 100% budget for every experiment. Which means, the total cost of annotate 1.37 objects per image for all the train set of the dataset.

Evaluation Metrics. For each annotation policy used to build the hybrid object detection dataset, we can experimentally get a budget-performance curve. The budget axis varies the percentage of the total cost D , and the model detection performance is taken to be mAP. We define the budget-aware metric as **Average mAP**, the normalized area under budget-mAP curve for the three budget ranges [10%, 30%], [30%, 50%], and [50%, 100%]. We have three to five data points for each range such that the metrics are less effected by noise and are able to compare different annotation policies in low, medium or high budget regimes. In the attachments, we show detailed experiment results, including budget-aware curves for each of the methods presented in this section.

Implementation Details. Our work fixes Faster-RCNN as the detection model, and develops the framework on top of it. However, this model can be replaced by any *off-the-shelf* object detector. Both teacher and student models use VGG16 as the backbone network pre-trained on ImageNet [25]. We follow the default set up in [40]. During training, the total number of epochs is set to 10, the learning rate is 0.01 for the first 8 epochs and 0.001 for the remaining. The batchsize is set to 16 on four-GPU cluster nodes equipped with Titan Xp. The student model is cloned from the ImageNet pretrained VGG16 for every active step and has the same training schedule as the teacher model. The source code of the framework will be made publicly available. In the LAL experiment, we collect 10 categories images from the MS-COCO dataset [12] to train the SVRs. Since the LAL method is dataset agnostic [53], the

training set categories are different from the 20 categories in PASCAL VOC.

B. Exploration on Image Side

In order to explore the influence of the uncertainty score on the model performance, we emulate the annotation process of the oracle using VOC07 and incorporate it into the active learning pipeline with only strong annotations and the FSOD method. We use three sampling methods RS, US, and LAL to build up the dataset for FSOD. We also report an experiment, in which we sample the images from the Smallest Uncertainty Score (SUS) to know whether to use the largest Uncertainty Score (US) images or the lowest ones.

Fig. 5: Budget-aware performance using strongly annotated images for three different sampling methods: **Orange bars** Random Sampling fully supervised (RS). **Blue bars** samples from most uncertain images (US). **Green Bars** samples from the least uncertain images (SUS). **Grey bars** samples from Learning Active Learning sampling (LAL).

Figure 5 compares three sampling methods in the active learning pipeline: random selection, largest and least uncertainty score, and LAL scores. We observe that the random sampling is a strong baseline in the object detection case and cannot be easily outperformed by only changing the sampling method. For the case of the uncertainty scores, when the budget is small (10%-30%), collecting images with small uncertainty score is more effective. However, when the dataset size is large enough, choosing large uncertainty scores is preferable. This behaviour follows the conclusions of Bengio *et. al.*[65], who suggest that it is better to train models using curriculum learning, *i.e.* giving *easy* images to the model in early training and then giving *hard* images once the model has learned better representations.

C. Advantages of Hybrid Training

In this section, we want to explore the influence of the annotation scheme on the model performance. We emulate the annotation process of the oracle using VOC07 into the active learning pipeline using strong and weak annotations with the

Fig. 6: Budget-aware performance using fully and hybrid training pipelines. **Orange bars** compare the two training pipelines using random sample while **Blue bars** show both pipelines using US sampling.

hybrid detection method. We use two sampling methods (RS and US) to construct the dataset for hybrid training.

Figure 9 shows the effectiveness of hybrid training. Both hybrid methods outperform their corresponding FSOD counterpart at every budget range. Specifically, when the budget range is small (under 50%), the hybrid training is clearly better than the fully supervised training.

D. Optimization Based Methods

In this part, we compare optimization based on US and optimization based on LAL against the hybrid baselines defined in Section IV-C. The optimization combines hybrid training and a sampling method at every step as specified in Section IV-D. Given that US optimization gives the best results, we refer to it as BAOD (budget aware object detection). Please keep in mind that it is the same BAOD shown in Figure 1.

Figure 7 indicates that the US and LAL optimization methods are the best ways to combine and use the hybrid annotations at every budget. Evidently, the optimization methods perform better than the other two in the three budget ranges. However, US is slightly better than LAL. Quantitatively, if we compare the budget that was used to achieve 95% of the full budget performance (*i.e.* 67.4% mAP), the hybrid random sampling requires 62% of VOC07 budget, whereas the US optimization method needs only 48.5% of it. This performance attests the effectiveness of this method. In the attachments, we include some visualizations from the weakly and strongly labeled sets at some budget levels for LAL.

Analysis of Different Budgets. The cost of strong annotations can vary due to varied complexity of the images and different methods. For instance, Papadopoulos *et al.* [4] created a method that reduces the time needed to draw bounding boxes to 7 seconds per instance. Here, we use that

Fig. 7: Budget-aware performance using FSOD, hybrid training, and optimization methods. **Blue bars** show US using FSOD. **Orange bars** show RS hybrid baseline. **Gold bars** show US Optimization (BAOD). **Dark Grey Bars** show LAL Optimization.

cost to display the behaviour of our method given a smaller gap between the cost of strongly and weakly annotated images. We compare the best FSOD sampling method against the RS hybrid baseline and the best optimization method.

Fig. 8: Budget-aware performance using a lower cost for strong annotations. **Blue Bars** represent FSOD with US sampling. **Orange Bars** represent Hybrid training with RS sampling. **Gold Bars** represent US Optimization. All the methods use a smaller gap between the weak and strong annotation costs.

In Figure 8, we observe that the use of optimization is still the best way to combine both annotations. We can see a smaller performance gap than before. The reason behind that behavior is that the difference between strong and weak cost is lower, thus the number of images that can be weakly annotated at the cost of one strong annotation is much smaller. Nonetheless, our method can still find a good balance to perform well with both kinds of annotations.

Fig. 9: Example of the images that are chosen by our best method to be annotated with strong and weak annotations for each active step. We can observe that at the beginning when the model is not good enough, the uncertainty score is not an accurate measure for the easiness of the image (Steps 2 and 4), as the model improves the Uncertainty becomes a better measure. From steps 6 to 10 we can observe that images with a lot of objects are strongly annotated while easy image with single objects and not too much background are weakly annotated.

E. Reinforcement Learning Experiments

At every active step, the Q agent gives insights on which images to label and which type of annotation is required. Different from the previous pipeline, the RL method does not evaluate all the unlabeled images, but renders a preference for easy, middle or hard images. This significantly expedites the training process. We firstly compute the number of required images from the budget and annotation types, then we stop evaluating unlabeled images' uncertainties, once we get enough images. Figure 10 shows that RL performs better than FSOD, since it learns a policy to weakly or strongly annotate images and takes advantage of hybrid training. Nevertheless, it performs worse than US and LAL optimization. The reinforcement learning environment is built by VOC12 training set and validation set. We simulate the active learning process on four Titan Xp GPUs for five days to train the Q-learning agent, and the agent is used to solve the budget aware object detection problem on VOC07 trainval set and test set.

F. Simulation of Unlimited Image Pool

We conduct this experiment to simulate a real-world application of our method. We combine VOC07 and VOC12 to simulate a larger pool of unlabelled images to choose from. The budget is set as the same amount needed to annotate VOC07. We report the results in Table I. If we train a detector on the union of both VOC07 and VOC12 trainval sets, it delivers a performance of 0.764 mAP. Table I shows that

Fig. 10: Budget-aware performance using Optimization based methods, FSOD and RL. **Blue bars** show the performance of FSOD using US. **Mint bars** show the performance of the RL pipeline. While **Grey** and **Yellow** bars show LAL and US Optimization (BAOD) respectively.

our method can reach 0.730 mAP using the same budget of VOC2007, which is a 2% mAP improvement over the model trained solely on VOC07. This result also elucidates that the proposed method (active hybrid training) can reach approximately 95.5% performance of using VOC0712, but with a third of the budget necessary (5,011) to strongly

annotate these two datasets (16,551).

TABLE I: Simulation of a Larger Unlabelled Image Pool

Train Set Pool	VOC07	VOC0712	VOC0712	VOC0712
Max Budget*	100%	87.2%	100%	330%
Final mAP	0.710	0.710	0.730	0.764

* The values are normalized by VOC07 trainval set cost.

V. CONCLUSION

In summary, we introduce a novel budget-aware perspective to study the unexplored dimensions of the object detection problem. With a fixed budget, we compare both optimization and learn based sample methods to build diverse hybrid supervised object detection datasets which consist of both image level supervision and instance level supervision. The evaluation of detectors that learned from these budget fixed datasets shows that the handcrafted optimization method on uncertainty score outperforms other general active learning method including random sampling, active learning, and reinforcement learning. With the optimal set-up, our proposed budget-aware approach can achieve the performance of a strongly supervised detector on Pascal-VOC 2007 while saving 12.8% of its original annotation budget. Furthermore, when 100% of the budget is used, our approach surpasses this performance by 2% mAP.

REFERENCES

- [1] R. Ranjan, V. M. Patel, and R. Chellappa, "Hyperface: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2018.
- [2] Y. Bai, Y. Zhang, M. Ding, and B. Ghanem, "Finding tiny faces in the wild with generative adversarial network," *CVPR*, 2018.
- [3] P. Hu and D. Ramanan, "Finding tiny faces," in *CVPR*. IEEE, 2017, pp. 1522–1530.
- [4] D. P. Papadopoulos, J. R. Uijlings, F. Keller, and V. Ferrari, "Extreme clicking for efficient object annotation," in *ICCV*. IEEE, 2017, pp. 4940–4949.
- [5] X. Du, M. El-Khamy, J. Lee, and L. Davis, "Fused dnn: A deep neural network fusion approach to fast and robust pedestrian detection," in *Applications of Computer Vision (WACV), 2017 IEEE Winter Conference on*. IEEE, 2017, pp. 953–961.
- [6] S. Wang, J. Cheng, H. Liu, and M. Tang, "Pcn: Part and context information for pedestrian detection with cnns," *arXiv preprint arXiv:1804.04483*, 2018.
- [7] P. Chattopadhyay, R. Vedantam, R. R. Selvaraju, D. Batra, and D. Parikh, "Counting everyday objects in everyday scenes," in *CVPR*, 2017, pp. 4428–4437.
- [8] D. Kang and A. Chan, "Crowd counting by adaptively fusing predictions from an image pyramid," *arXiv preprint arXiv:1805.06115*, 2018.
- [9] J. Collomosse, T. Bui, M. J. Wilber, C. Fang, and H. Jin, "Sketching with style: Visual search with sketches and aesthetic context," in *ICCV*, 2017, pp. 2679–2687.
- [10] C. Mu, J. Zhao, G. Yang, J. Zhang, and Z. Yan, "Towards practical visual search engine within elasticsearch," *arXiv preprint arXiv:1806.08896*, 2018.
- [11] D. Mahajan, R. Girshick, V. Ramanathan, K. He, M. Paluri, Y. Li, A. Bharambe, and L. van der Maaten, "Exploring the limits of weakly supervised pretraining," *arXiv preprint arXiv:1805.00932*, 2018.
- [12] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *ECCV*. Springer, 2014, pp. 740–755.
- [13] K. Konyushkova, J. Uijlings, C. H. Lampert, and V. Ferrari, "Learning intelligent dialogs for bounding box annotation," in *CVPR*, 2018.
- [14] D. P. Papadopoulos, J. R. Uijlings, F. Keller, and V. Ferrari, "We don't need no bounding-boxes: Training object class detectors using only human verification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 854–863.
- [15] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *CVPR*, 2016, pp. 779–788.
- [16] J. Redmon and A. Farhadi, "Yolo9000: Better, faster, stronger," *arXiv preprint arXiv:1612.08242*, 2016.
- [17] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," *CoRR*, vol. abs/1311.2524, 2013. [Online]. Available: <http://arxiv.org/abs/1311.2524>
- [18] R. Girshick, "Fast r-cnn," in *ICCV*, 2015, pp. 1440–1448.
- [19] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: object detection via region-based fully convolutional networks," *CoRR*, vol. abs/1605.06409, 2016. [Online]. Available: <http://arxiv.org/abs/1605.06409>
- [20] 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 (NIPS)*, 2015.
- [21] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," *arXiv preprint arXiv:1703.06870*, 2017.
- [22] T. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie, "Feature pyramid networks for object detection," *CoRR*, vol. abs/1612.03144, 2016. [Online]. Available: <http://arxiv.org/abs/1612.03144>
- [23] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," in *Proceedings of the IEEE*, 1998, pp. 2278–2324.
- [24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778.
- [25] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, *et al.*, "Imagenet large scale visual recognition challenge," *IJCV*, vol. 115, no. 3, pp. 211–252, 2015.
- [26] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *IJCV*, vol. 88, no. 2, pp. 303–338, 2010.
- [27] H. Su, J. Deng, and L. Fei-Fei, "Crowdsourcing annotations for visual object detection," in *Workshops at the Twenty-Sixth AAAI Conference on Artificial Intelligence*, vol. 1, 2012.
- [28] H. Bilen, M. Pedersoli, and T. Tuytelaars, "Weakly supervised object detection with convex clustering," in *CVPR*, 2015, pp. 1081–1089.
- [29] R. G. Cinbis, J. Verbeek, and C. Schmid, "Weakly supervised object localization with multi-fold multiple instance learning," *TPAMI*, vol. 39, no. 1, pp. 189–203, 2017.
- [30] V. Kantorov, M. Oquab, M. Cho, and I. Laptev, "Contextlocnet: Context-aware deep network models for weakly supervised localization," in *ECCV*. Springer, 2016, pp. 350–365.
- [31] H. Bilen and A. Vedaldi, "Weakly supervised deep detection networks," in *CVPR*, 2016, pp. 2846–2854.
- [32] B. Zhuang, L. Liu, Y. Li, C. Shen, and I. Reid, "Attend in groups: a weakly-supervised deep learning framework for learning from web data," *arXiv preprint arXiv:1611.09960*, 2016.
- [33] R. Gokberk Cinbis, J. Verbeek, and C. Schmid, "Multi-fold mil training for weakly supervised object localization," in *CVPR*, 2014, pp. 2409–2416.
- [34] D. Li, J.-B. Huang, Y. Li, S. Wang, and M.-H. Yang, "Weakly supervised object localization with progressive domain adaptation," in *CVPR*, 2016, pp. 3512–3520.
- [35] J. Hoffman, D. Pathak, T. Darrell, and K. Saenko, "Detector discovery in the wild: Joint multiple instance and representation learning," in *CVPR*, 2015, pp. 2883–2891.
- [36] M. Roohan and Y. Wang, "Weakly supervised localization of novel objects using appearance transfer," in *CVPR*, 2015, pp. 4315–4324.
- [37] C. Wang, W. Ren, K. Huang, and T. Tan, "Weakly supervised object localization with latent category learning," in *ECCV*. Springer, 2014, pp. 431–445.
- [38] Z. Jie, Y. Wei, X. Jin, J. Feng, and W. Liu, "Deep self-taught learning for weakly supervised object localization," *arXiv preprint arXiv:1704.05188*, 2017.
- [39] Y. Zhang, Y. Bai, M. Ding, Y. Li, and B. Ghanem, "W2f: A weakly-supervised to fully-supervised framework for object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 928–936.
- [40] J. Yang, J. Lu, D. Batra, and D. Parikh, "A faster pytorch implementation of faster r-cnn," <https://github.com/jwyang/faster-rcnn.pytorch>, 2017.

- [41] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *ECCV*. Springer, 2016, pp. 21–37.
- [42] H. O. Song, Y. J. Lee, S. Jegelka, and T. Darrell, "Weakly-supervised discovery of visual pattern configurations," in *Advances in Neural Information Processing Systems (NIPS)*, 2014, pp. 1637–1645.
- [43] P. Tang, X. Wang, X. Bai, and W. Liu, "Multiple instance detection network with online instance classifier refinement," *CoRR*, vol. abs/1704.00138, 2017. [Online]. Available: <http://arxiv.org/abs/1704.00138>
- [44] L.-J. Li and L. Fei-Fei, "Optimol: automatic online picture collection via incremental model learning," *International journal of computer vision*, vol. 88, no. 2, pp. 147–168, 2010.
- [45] C. Rosenberg, M. Hebert, and H. Schneiderman, "Semi-supervised self-training of object detection models." in *WACV/MOTION*, 2005, pp. 29–36.
- [46] X. Chen, A. Shrivastava, and A. Gupta, "Neil: Extracting visual knowledge from web data," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1409–1416.
- [47] S. Laine and T. Aila, "Temporal ensembling for semi-supervised learning," *arXiv preprint arXiv:1610.02242*, 2016.
- [48] I. Radosavovic, P. Dollár, R. Girshick, G. Gkioxari, and K. He, "Data distillation: Towards omni-supervised learning," *arXiv preprint arXiv:1712.04440*, 2017.
- [49] R. Hu, P. Dollár, K. He, T. Darrell, and R. Girshick, "Learning to segment every thing," *arXiv preprint*, 2018.
- [50] G. Chéron, J.-B. Alayrac, I. Laptev, and C. Schmid, "A flexible model for training action localization with varying levels of supervision," *arXiv preprint arXiv:1806.11328*, 2018.
- [51] G.-J. Qi, X.-S. Hua, Y. Rui, J. Tang, and H.-J. Zhang, "Two-dimensional active learning for image classification," in *CVPR*. IEEE, 2008, pp. 1–8.
- [52] W. H. Beluch, T. Genewein, A. Nürnberger, and J. M. Köhler, "The power of ensembles for active learning in image classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 9368–9377.
- [53] K. Konyushkova, R. Sznitman, and P. Fua, "Learning active learning from data," in *Advances in Neural Information Processing Systems*, 2017, pp. 4225–4235.
- [54] A. Kapoor, K. Grauman, R. Urtasun, and T. Darrell, "Active learning with gaussian processes for object categorization," in *ICCV*. IEEE, 2007, pp. 1–8.
- [55] J.-J. Zhu and J. Bento, "Generative adversarial active learning," *arXiv preprint arXiv:1702.07956*, 2017.
- [56] M. Hasan and A. K. Roy-Chowdhury, "Context aware active learning of activity recognition models," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 4543–4551.
- [57] F. C. Heilbron, J.-Y. Lee, H. Jin, and B. Ghanem, "What do i annotate next? an empirical study of active learning for action localization," in *ECCV*, 2018.
- [58] K. Konyushkova, R. Sznitman, and P. Fua, "Introducing geometry in active learning for image segmentation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 2974–2982.
- [59] C.-C. Kao, T.-Y. Lee, P. Sen, and M.-Y. Liu, "Localization-aware active learning for object detection," *arXiv preprint arXiv:1801.05124*, 2018.
- [60] B. Liu and V. Ferrari, "Active learning for human pose estimation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 4363–4372.
- [61] I. Misra, R. Girshick, R. Fergus, M. Hebert, A. Gupta, and L. van der Maaten, "Learning by asking questions," *arXiv preprint arXiv:1712.01238*, 2017.
- [62] D. Wang and Y. Shang, "A new active labeling method for deep learning," *2014 International Joint Conference on Neural Networks (IJCNN)*, pp. 112–119, 2014.
- [63] D. D. Lewis and J. Catlett, "Heterogeneous uncertainty sampling for supervised learning," in *Machine Learning Proceedings 1994*. Elsevier, 1994, pp. 148–156.
- [64] C. J. Watkins and P. Dayan, "Q-learning," *Machine learning*, vol. 8, no. 3-4, pp. 279–292, 1992.
- [65] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *Proceedings of the 26th annual international conference on machine learning*. ACM, 2009, pp. 41–48.