Data-efficient Weakly-supervised Learning for On-line Object Detection under Domain Shift in Robotics

Elisa Maiettini¹, Raffaello Camoriano², Giulia Pasquale¹, Vadim Tikhanoff³, Lorenzo Rosasco²⁴⁵ and Lorenzo Natale¹

Abstract—Several object detection methods have recently been proposed in the literature, the vast majority based on Deep Convolutional Neural Networks (DCNNs). Such architectures have been shown to achieve remarkable performance, at the cost of computationally expensive batch training and extensive labeling. These methods have important limitations for robotics: Learning solely on off-line data may introduce biases (the so-called domain shift), and prevents adaptation to novel tasks.

In this work, we investigate how weakly-supervised learning can cope with these problems. We compare several techniques for weakly-supervised learning in detection pipelines to reduce model (re)training costs without compromising accuracy. In particular, we show that diversity sampling for constructing active learning queries and strong positives selection for self-supervised learning enable significant annotation savings and improve domain shift adaptation. By integrating our strategies into a hybrid DCNN/FALKON on-line detection pipeline [1], our method is able to be trained and updated efficiently with few labels, overcoming limitations of previous work. We experimentally validate and benchmark our method on challenging robotic object detection tasks under domain shift.

I. INTRODUCTION

Object detection is a fundamental problem in computer vision and robotics. Differently from object classification, which deals with learning a mapping from a whole image to object labels, object detectors both locate and classify one or more target objects in an image.

State-of-the-art methods in Computer Vision tackle this problem by employing Deep Convolutional Neural Networks (DCNNs) for feature extraction, region proposal and classification. These approaches determined a breakthrough in terms of accuracy, especially when the learning can be fully supervised. However, they are constrained by demanding training times and large annotated datasets. Such limitations are particularly significant for object detection in robotics, which requires efficient model training and adaptation to novel tasks and scenarios with limited expert labeling.

Previous works in robotics propose to address the annotated data requirements by collecting automatically labeled images, e.g., by exploiting a human teacher and 3D information in a constrained scenario [2], [3]. Moreover, long training times have been tackled in [4] and [5] with the introduction of a hybrid on-line detection pipeline, which leverages on a DCNN feature extractor and an efficient Kernel-based classifier. While allowing for a natural teacher-learner interaction and accurate object detection, this approach supports limited generalization to novel, unseen, scenarios [3], [1].

Robots can acquire training images autonomously during operation. Such images come in streams and can carry useful information, eventually containing the objects of interest, but they are not labeled. Promising results have been achieved by integrating the aforementioned on-line object detector with weakly-supervised learning (WSL) [6], which targets learning from partially-annotated datasets. The resulting pipeline exploits the labeled part of the dataset to train an initial seed model, which is then used to process the unlabeled images. The obtained predictions are used to form “hard” and “easy” image sets. The former is used to construct queries for human annotation (Active Learning (AL) framework [7]), while the second is added to the training set for Self-supervised Learning (SSL) [8], [9]. While showing promising results, this method presents some limitations. Firstly, the unsupervised data processing is pool-based [7], that is, all unlabeled images are evaluated before query selection. This is not suitable for robotic agents, which cannot wait the end of the acquisition to request annotations, but need to do it interactively during exploration. Therefore, they should process images frame by frame and make individual query decisions in a stream-based fashion [7]. This latter strategy is more suited to robotics, but might yield to lower accuracy since queries are constructed using limited information on the unlabeled set [7]. Moreover, the on-line method presented in [1] iterates multiple times over the unlabeled data, which, while allowing to refine the data selection, slows down learning. In this paper, we focus on the stream-based scenario with the aim of increasing the human labeling efficiency of weakly-supervised on-line object detection. Moreover, we consider the case where only one pass over the unlabeled data is allowed. The main contributions of this work are:

- We present and empirically evaluate several AL techniques for detection. We compare pool-based and stream-based AL in a challenging robotic scenario and propose a solution to overcome limitations of the latter.
- We investigate domain shift effects on AL and SSL and how wrongly self-annotated data can degrade accuracy.
- Finally, we propose an SSL sampling method to overcome this problem and we empirically demonstrate how SSL can effectively reduce labeling efforts.
This paper is organized as follows: we introduce object detection, and weakly-supervised learning (Sec. II) and we cover related work (Sec. III). In Sec. IV, we present our efficient detection methods, which are analyzed and validated in Sec. V. Sec. VI concludes the paper.

II. BACKGROUND

A. Object Detection

Given an image $x \in \mathcal{I}$, the goal of object detection is to both localize and recognize, in $x$, all the objects belonging to a specified set of $T$ classes, $C$. We define $f_{det} : \mathcal{I} \rightarrow \mathcal{D}$ as the ground truth associating $x$ with the $N_x \in \mathbb{N}_1$ pairs of class labels $y^* \in C$ and locations $z^* \in \mathcal{Z}$ of all the target objects it contains. The aim of object detection is to construct a function (detector) $\hat{f}_{det} : \mathcal{I} \rightarrow \mathcal{D}$ approximating $f_{det}$ as closely as possible according to a specified quality metric. In this work, we consider confidence-based detectors $f_{det} : \mathcal{I} \rightarrow \mathcal{D} \times \mathbb{R}$, which associate a confidence score $s_i \in \mathbb{R}$ to each predicted output pair:

$$\hat{f}_{det}(x) = \{(\hat{y}_i, \hat{z}_i, s_i)\}_{i=1}^{N_x}.$$  

Thresholding $s_i$ with a parameter $\tau \in \mathbb{R}$ allows to select predictions according to the desired trade-off between precision and recall, where $\tau$ is the minimum required confidence score for the detector to predict $(\hat{y}_i, \hat{z}_i)$. 

B. From Fully- to Weakly-supervised Learning

The supervised learning approach to object detection is centered on learning the detector $f_{det}$ from an annotated (supervised) dataset $S_n = \{(x_i, Y_i)\}_{i=1}^{n}$ of images and corresponding annotations $Y \in \mathcal{D}$. The methods described in Sec. III-A fall in this category. They contributed to a clear progress in detection accuracy and prediction speed, however, they need expensively-annotated large-scale datasets to be optimized. This characteristic does not meet the robotic requirement for a detector to adapt to a variety of tasks, potentially unknown a-priori, in a short time span. However, while large annotated datasets might not be available, plenty of unsupervised images are usually accessible to robots. In this context, a training set $S_n = L \cup U$ is typically composed of a labeled subset $L = \{(x_i, Y_i)\}_{i=1}^{L}$ and an unlabeled subset $U = \{x_i\}_{i=1}^{U}$. WSL allows the agent to select unsupervised images from $U$ and acquire their labels semi-autonomously for updating the detector, minimizing human effort and improving accuracy. WSL includes several subclasses of methods, depending on the label-acquisition mechanism [8], [9]. The most relevant for this work are Active Learning and Self-supervised Learning.

Active Learning. AL [7] interactively queries unsupervised examples for expert labeling to minimize human annotation and maximize accuracy. Unlabeled examples are chosen from $U$ according to a scoring function and a sampling strategy. Their labels are then queried to an expert, and newly-annotated examples are added to $L$ for training. If all images in $U$ are accessible at selection time, sampling is referred to as pool-based. Otherwise, if only one candidate from $U$ is accessible, sampling becomes a binary decision on keeping or dropping it and is called stream-based. The AL selection criterion we focus on is uncertainty sampling, which picks the examples the model is least confident about.

Self-supervised Learning. In SSL, unlabeled images are annotated by the detector itself with no human intervention, propagating predicted labels to high-confidence regions of the input space by exploiting the geometry of the input data distribution. This technique is effective if the detector is not overconfident of its predictions and if the confidence threshold for propagating predicted labels is strict enough.

III. RELATED WORK

A. Classic Object Detection Methods

Early approaches to object detection were based on feature dictionaries [10] or specific kinds of image descriptors [11], [12], [13]. Feature vectors were separately classified by supervised learning methods. Despite yielding limited accuracy, these approaches had the advantage of being parsimonious in terms of computations and dataset size.

More recently, object detection experienced significant progress thanks to the introduction of DCNN-based methods. This determined clear improvements in terms of predictive performance, mainly due to the powerful representation capabilities of deep networks. Such approaches include two-stage detectors based on Region Proposal Networks (RPNs) (like e.g. Faster R-CNN [14] and Mask R-CNN [15]) and related extensions [16], [17], [18], [19]. These methods employ a DCNN to perform (i) region candidates predictions, (ii) per extensions [16], [17], [18], [19]. These methods employ a DCNN to perform (i) region candidates predictions, (ii) per region feature extraction and (iii) region classification and refinement.

Alternative end-to-end approaches include one-stage detectors, which replace the RPN with a fixed, dense grid of candidate bounding boxes. One such example is SSD [20], achieving accuracies competitive with the RPN-based Faster R-CNN and high frame rate. Another one-stage method, RetinaNet [21], rebalances foreground and background examples through the so-called Focal Loss.

B. Efficient Object Detection for Robotics

Despite their high accuracy, the described approaches typically require (i) long optimization time and (ii) large-scale annotated datasets for adaptation to novel tasks. These aspects limit their adoption in robotics.

Computational efficiency. A well-known issue of DCNN-based pipelines is that they suffer from catastrophic forgetting when optimized on new data [22], [23]. This limitation implies retraining these models on the full dataset, causing long adaptation time. To address this issue, a recent work for robotic object detection leverages fast classifiers to enable on-line adaptation [4], [5]. It has been shown [5] that efficient multi-stage pipelines can be constructed by combining DCNN-based RPNs and feature extractors (namely, based on Faster R-CNN or Mask R-CNN) with
large-scale Kernel classifiers [24], [25], [26]. According to this approach, the DCNN is pre-trained off-line on a large representative dataset, yielding a powerful and transferable learned representation, which is kept fixed during on-line operation. The actual regions classification is performed by the integration of an efficient hard-negatives bootstrapping approach (the Minibootstrap [5]) with a set of FALKON classifiers [24], [25].

**Labeling efficiency.** Labeling efficiency is another key requirement for robotic object detection. The broad class of WSL methods [8], [9] provides a rich set of tools towards this goal, in particular AL and SSL – introduced in Sec. [I]

After successful applications to deep object classification [27], [28], [29], [30], AL has been recently applied also to object detection [31], [32], [33], [34]. For instance, recently, detection-specific image scoring functions (like e.g., localization tightness and stability [35]), have been proposed. Similarly to AL, SSL has been recently applied to object detection. For instance, in [36], SSL is employed for dataset augmentation and training object detectors. Moreover, the authors point out that vanilla SSL can degrade accuracy in presence of domain shift. We also observed the same issue and we propose a simple yet effective solution in Sec [V-D]

Recent approaches integrate both AL and SSL techniques into the same detection pipeline, like e.g., the Self-supervised Sample Mining [6], [37] (SSM). SSM sorts unsupervised images into separate candidate sets for further AL and SSL processing, according to the predictive confidence scores of the underlying DCNN based detection model [16]. Furthermore, in [1], SSM is extended to enable on-line adaptive object detection for robotics, by integrating the WSL sample selection strategy with the on-line object detection method [4], [5]. While in [1], the WSL strategy is kept unchanged with respect to SSM, in this work, we present an empirical analysis of different AL techniques in a challenging robotic scenario, proposing a solution for the SSL failure cases under domain shift, and improving overall labeling efficiency.

IV. METHODS

In this work, a robot is asked to detect a set of object instances in an unconstrained environment (referred to as TARGET-TASK). The detection system is initialized with a set of convolutional weights, previously trained off-line on a separate set of objects (referred to as FEATURE-TASK). A first detection model is trained during a brief interaction with a human, in a constrained scenario (the TARGET-TASK-LABELED). Then, the robot autonomously explores the environment, acquiring a stream of images in a new setting. These images are not labeled (TARGET-TASK-UNLABELED) and are used to adapt the detector on-line. In the next sections, we present the proposed pipeline (Sec. [IV-A]) and the learning protocol (Sec. [IV-B]). Then, we present all the considered AL and SS techniques and the proposed approaches (Sec. [IV-C] and [IV-D], respectively).

Fig. 1. Overview of the proposed pipeline. Refer to Sec. [IV-A] for details.

A. Pipeline Description

The proposed WSL pipeline (see Fig. 1) is composed of four main components: (i) the On-line Object Detection, (ii) the Scoring function, (iii) the AL Selection policy, and (iv) the SS Selection policy.

**On-line Object Detection (OOD).** For this module, we follow the method proposed in [5], but considering the implementation presented in [38], [39]. This is an on-line learning approach consisting of two stages: (i) region proposals and feature extraction, and (ii) region classification and bounding-box refinement. The first stage relies on layers from Mask R-CNN [15] (specifically, the convolutional layers, the RPN [40] and the RoI Align layer [15]). In particular, this part is used to extract a set of Regions of Interest (RoIs) from an image and encode them into a set of features. The second stage is composed of a set of FALKON [41] binary classifiers (one for each class of the TARGET-TASK) and Regularized Least Squares (RLS) [42], respectively for the classification and the refinement of the proposed RoIs. Classifiers are trained with an approximate bootstrapping approach, called Minibootstrap [5], which addresses the well-known issue of background-foreground class imbalance in object detection, while maintaining a short training time.

**Scoring function.** This function assigns a confidence score to the predictions for the TARGET-TASK-UNLABELED images. This score is then used by the AL and SS Selection policies to decide which images need to be manually annotated or can be considered as pseudo ground truth. For this part, we employ the Cross-Image Validation (CIV) proposed in SSM [6]. CIV stitches predicted image patches from the TARGET-TASK-UNLABELED on random images, sampled from TARGET-TASK-LABELED. Then, it executes the detector on the stitched images and computes...
a consistency score from the obtained confidence scores [6].

**AL and SS Selection policies.** Given the predicted detections obtained by the OOD and the consistency score extracted by the CIV, these two policies decide whether an image of the TARGET-TASK-UNLABELED is queried for annotation or the predicted detections are confident enough to be used for self-supervision. Note that, while the first two components of the pipeline remain unchanged with respect to [1], our main contribution relies on these last two. Specifically, for the AL Selection policy, we replace the coin-flipping AL strategy, used in [1] and [6], with several AL techniques, comparing their performance. The adopted AL baselines and the proposed solution are listed in Sec. [IV-C]. Instead, for the SS Selection policy, we replace the pool-based technique used in [1], [6] (i.e., a top-k strategy using the CIV consistency score for ranking) with a stream-based baseline and a novel strategy, both described in Sec. [IV-D].

Finally, another major difference with respect to [1], is that we consider the case where only one pass over the TARGET-TASK-UNLABELED is allowed, while [1] presented an iterative process. This aspect is crucial for speeding up WSL. However, it makes detector refinement more challenging.

**B. Learning Protocol**

The learning process is divided into: (i) **Supervised phase**, and (ii) **Weakly-supervised phase** (see Fig. 1). The first is performed within a few seconds of interaction with a human on the TARGET-TASK-LABELED, yielding a first detection model (the seed model). The Mask R-CNN's features are used for training the FALKON classifiers and the RLS regressors (see [5] for details on the on-line training). Then, in the WSL phase, the SS pseudo ground truth and AL queries are selected from the TARGET-TASK-UNLABELED as described in Sec. [IV-A], using the seed model's confidence scores. Finally, the on-line detector is retrained on the new dataset.

**C. Active Learning Strategies**

For AL selection, we considered both (i) Stream-based approaches, which are the focus of this work, being suited to robotic scenarios, and (ii) Pool-based ones, which provide an upper bound on accuracy. A simple, yet often effective, pool-based strategy is to sample uniformly at random the images with a confidence score below a threshold (Uniform random in Sec. [V]). Another diversity sampling strategy is to execute k-means clustering [42] on the image-level features and select the resulting cluster centers (K-means-based AL in Sec. [V]). In our analysis, we report results for both baselines. In stream-based AL settings, a simple selection strategy involves confidence score thresholding followed by coin flipping [6], for implementing uncertainty and diversity sampling, respectively (coin-flip AL in Sec. [V]). Another solution is to exploit temporal coherence in image sequences to enforce sampling diversity [43]. To this aim, we consider the Fixed temporal window strategy, which employs a temporal window of fixed size $\Delta$ so that if frame $t$ is selected, any other frame within $[t - \Delta, t + \Delta]$ is no longer considered. While enforcing diversity, this strategy, by fixing $\Delta$, does not take into account: (i) the exploration duration, that is, the size $n_U$ of TARGET-TASK-UNLABELED, which might be known a-priori even in stream-based scenarios, and (ii) the available manual annotation budget $k$. We show in Sec. [V-C] that this results in poor performance for low $k$ when the TARGET-TASK-UNLABELED is redundant. For this reason, in this work, we propose to use an adaptive temporal window, defined as $\Delta_{n_U,k} = \frac{k}{2U}$ (Adaptive temporal window in Sec. [V]).

**D. Self-supervised Learning Strategies**

For SS selection, we consider two stream-based baselines. The first is the SS baseline, which selects all the images passing CIV as pseudo ground truth. However, we show in Sec. [V-D] that under domain shift this leads to model degradation due to the abundance of false negatives. For this reason, in this work, we propose a more conservative strategy, namely SS positives only, which only selects positive predictions and leaves out negative ones. In Sec. [V-D] we show that integrating SS positives only with AL further increases accuracy with the same labeling budget.

**V. EXPERIMENTS**

We now report on our experimental analysis, including AL techniques comparison, SSL performance evaluation, and overall AL+SSL pipeline accuracy study under domain shift.

**A. Datasets Description**

We consider two robotic datasets in our empirical evaluation. The first is iCubWorld Transformations [44] (iCWT), which contains images for 200 objects belonging to 20 categories (10 instances for each category). iCWT contains images of hand-held objects, demonstrated by a human teacher to the robot. Each object is acquired with different sequences representing specific viewpoint transformations: 2D rotation (2D ROT), generic rotation (3D ROT), translation (TRANSL), scaling (SCALE) and all transformations...
randomly combined (MIX) (see [44]). The second dataset is a collection of table-top sequences [1] (TABLE-TOP), depicting 21 of the 200 objects from iCWT randomly placed on a table with two different tablecloths: (i) pink/white pois (POIS) and (ii) white (WHITE). Refer to [1] for further details about the dataset. Both datasets are publicly available. The two datasets contain the same objects, but with an important domain shift: iCWT frames include the hand of the teacher, whereas TABLE-TOP has different backgrounds and depicts objects on a table.

B. Experimental Setup

To evaluate the proposed approaches, we consider the scenario of a robot relying on a detector previously trained with human interaction on a set of objects (Supervised phase in Sec. [IV-B]). This model needs to be refined in order to generalize to a different setting (i.e., a table top) by exploiting the unlabeled data collected by the robot during autonomous exploration (Weakly-supervised phase in Sec. [IV-B]).

In the presented experimental analysis we consider as FEATURE-TASK (see Sec. [IV]) the general purpose computer vision dataset MS COCO [45]. The weights of the Feature extractor are learned by training Mask R-CNN on the FEATURE-TASK (ResNet50 [46] has been considered as Mask R-CNN’s convolutional backbone). For this purpose, we use the available pre-trained Mask R-CNN weights.

In our experiments, autonomous exploration is simulated by means of the two robotic datasets described in Sec. [V]. Specifically, for the Supervised phase we employ a subset of the iCWT, considering 21 of the total 200 objects. All the transformations, except from MIX, are considered, resulting in a TARGET-TASK-LABELED of size \( n_L \sim 6K \). Instead, for the Weakly-supervised phase, we consider the WHITE sequence from TABLE-TOP as TARGET-TASK-UNLABELED and the POIS sequence as test set to evaluate performance, respectively of size \( \sim 2K \) and \( \sim 1K \). Note that, when implementing AL queries on the TARGET-TASK-UNLABELED, we simulate human annotation for AL by fetching the actual ground truth from the dataset.

We report performance in terms of mAP (mean Average Precision) at the IoU (Intersection over Union) threshold set to 0.5, as defined for Pascal VOC 2007 (see [47]). Specifically, we repeat each experiment for three trials and we present the results, reporting the mean and the standard deviation of the obtained accuracy.

C. Active Learning Sampling Strategies Comparison

In this section, we compare the AL techniques described in Sec. [IV] considering different manual annotation budgets. To this aim, we report in Fig. 2 the mAP trend obtained by increasing the AL query budget during the Weakly-supervised phase. Specifically, we report in red shades the performance obtained by the pool-based strategies (namely, k-means-based AL and Uniform random from Sec. [IV]), and in green shades the stream-based ones (namely, Coin-flip AL, Fixed temporal window, and Adaptive temporal window from Sec. [IV]). We set the fixed temporal window size as \( \Delta = 6 \).

As it can be observed in Fig. 2 as expected, pool-based methods achieve the best mAP trends. We consider them as the upper bounds of stream-based approaches. Notably, we observe that the Uniform random baseline is almost as effective as \( k \)-means based-AL and they both present an early steep slope for limited manual annotation budgets. These two aspects are due to the fact that the considered TABLE-TOP dataset, simulating an autonomous robot’s exploration, contains sequences of similar (and thus redundant) frames which need to be properly filtered during data selection. This aspect of the dataset is also the main cause for the poor performance obtained by the two stream-based techniques: Coin-flip AL and Fixed temporal window. Indeed, while being more suited for a robotic application, by reasoning only on a frame-by-frame fashion, they lack global information on the whole data distribution, which turns out to be a critical drawback especially for limited manual annotation budgets. However, for higher budgets, the Fixed temporal window baseline achieves accuracies closer to the pool-based upper bounds. It is important to note that the proposed Adaptive temporal window stream-based approach achieves significantly higher mAP values for low annotation budgets, being the most successful stream-based approach in such regime and closely matching pool-based ones.

D. Self-supervised Learning Evaluation

In this section, we investigate the impact of domain shift on SSL. To this aim, we report in Tab. 1 the results of applying the SS baseline (as defined in Sec. [IV]) in the two following scenarios:

- **Large domain shift.** In this case (table-top row in Tab. 1), we consider the scenario in which the TARGET-TASK-UNLABELED presents a completely different setting (i.e., a table top) with respect to the TARGET-TASK-LABELED (i.e., hand-held). To this end we used the two datasets described in Sec. [V-B].
- **Small domain shift.** In this case (icwt row in Tab. 1), TARGET-TASK-LABELED and TARGET-TASK-UNLABELED present similar conditions. The only difference in the latter one is that the objects are presented, unlabeled, with different view poses.

| Domain Shift | Supervised phase (mAP(\%)) | SS baseline (mAP(\%)) | SS samples percentage |
|----------------|---------------------------|------------------------|----------------------|
| table-top       | 48\%                      | 37\%                   | ~12\%                |
| icwt            | 41\%                      | 47\%                   | ~35\%                |

### TABLE I

**SS baseline results for large (1st row) and small (2nd row) domain shift between supervised and unsupervised datasets.**
To this aim, we considered as TARGET-TASK a 30-object identification task from iCWT. For each object, we then use the TRANSL sequence (~2K images) as TARGET-TASK-LABELED and the union of the 2D ROT, 3D ROT, and SCALE sequences (~6K images) as the TARGET-TASK-UNLABELED. We test on the MIX sequences of all the objects (~4.5K images).

Tab. I reports the results obtained in both cases. For each row, we report the mAP after the **Supervised phase** (first column) and after the **Weakly-supervised phase** (second column). Moreover, in the third column we report the average percentage of samples selected by the SS process over the total. As it can be observed, adding self-supervised data, with small domain shift, results in an improvement in accuracy. On the contrary, with a larger domain shift, it leads to a significant accuracy deterioration. A reason for this phenomenon can be identified by analyzing the pseudo ground truth generated by the SS process. We report in Fig. 3 some representative images depicting in green the region proposal candidates classified as background by the detection system and that are therefore added as negative samples to the dataset by the SS baseline. The actual detections which instead are considered as positive samples in the SSL process are shown in red. It can be noticed that, with large domain shift, only few objects are correctly detected and therefore added to the training set as positives, while most others are false negatives which are automatically annotated as background samples. Clearly, retraining the detection model with such poorly-labeled dataset leads to the sharp performance decay shown in Tab. I. We propose a solution to this problem by introducing the SS **positives only** strategy described in Sec. IV. According to this more conservative strategy, only the detections predicted as positive are included in the self-supervised dataset. The other regions are filtered away, since they might contain numerous false negatives (i.e., objects of interest wrongly classified as background by the current model). We report the results obtained with the proposed approach in Tab. II. The proposed strategy effectively removes wrong labels from the dataset, successfully yielding a significant improvement in performance instead of a two-digits decay.

**E. Overall On-line Detection Pipeline Performances**

As a final analysis, we demonstrate that by combining one of the AL techniques presented in Sec. [IV] with the proposed SS **positives only** strategy, it is possible to achieve further savings in manual labeling. To this aim, in Fig. 4 we compare vanilla Coin-flip AL (dark green) with the integration of Coin-flip AL with our SS **positives only** SSL strategy (light green). As it can be noticed, adding the samples chosen by the proposed SS strategy allows to obtain a significantly higher mAP without increasing the number of manually-annotated images.

**VI. CONCLUSIONS**

In this paper, we present and empirically evaluate a stream-based weakly-supervised on-line object detection pipeline for robotics, improving on previous work [1]. We compare several AL and SS techniques, proposing solutions to overcome stream-based AL limitations and performance degradation in SSL due to domain shift. Reliable perception and efficient adaptation to novel tasks and conditions are prominent skills for robots to operate in ever-changing environments. Therefore, fast optimization times and annotated training data efficiency are critical aspects for learning-based object detection systems in robotics. From this perspective, this work proposes novel solutions to significantly alleviate the annotation burden for on-line model adaptation to novel settings while maximizing accuracy, for enabling robots to explore the environment more autonomously.
[42] T. Hastie, R. Tibshirani, and J. Friedman, *The elements of statistical
learning: data mining, inference, and prediction.* Springer Science
& Business Media, 2009.

[43] H. H. Aghdam, A. Gonzalez-Garcia, J. v. d. Weijer, and A. M. Lopez,
“Active learning for deep detection neural networks,” in *Proceedings of
the IEEE/CVF International Conference on Computer Vision (ICCV),
*October 2019.

[44] G. Pasquale, C. Ciliberto, F. Odone, L. Rosasco, and L. Natale,
“Are we done with object recognition? the icub robot’s perspective,”
*Robotics and Autonomous Systems*, vol. 112, pp. 260 – 281, 2019.
[Online]. Available: http://www.sciencedirect.com/science/article/pii/
S0921889018300332

[45] 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 *European Conference on Computer Vision (ECCV),
*Zürich, 2014, oral. [Online]. Available: /se3/wp-content/uploads/
2014/09/coco_eccv.pdf, http://mscoco.org

[46] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for
image recognition,” *arXiv preprint arXiv:1512.03385*, 2015.

[47] 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, June 2010. [Online]. Available: https:
//doi.org/10.1007/s11263-009-0275-4