Incremental Domain Adaptation of Deformable Part-based Models

Jiaolong Xu[1,2]
jiaolong@cvc.uab.es
Sebastian Ramos[1,2]
sramosp@cvc.uab.es
David Vázquez[1]
dvazquez@cvc.uab.es
Antonio M. López[1,2]
antonio@cvc.uab.es

[1] Computer Vision Center
Edifici O, Campus UAB, 08193
Bellaterra (Barcelona), Spain

[2] Computer Science Dept.
Universitat Autònoma de Barcelona
Campus UAB, Bellaterra (Barcelona), Spain

Abstract

Nowadays, classifiers play a core role in many computer vision tasks. The underlying assumption for learning classifiers is that the training set and the deployment environment (testing) follow the same probability distribution regarding the features used by the classifiers. However, in practice, there are different reasons that can break this constancy assumption. Accordingly, reusing existing classifiers by *adapting* them from the previous training environment (*source domain*) to the new testing one (*target domain*) is an approach with increasing acceptance in the computer vision community. In this paper we focus on the *domain adaptation* of deformable part-based models (DPMs) for object detection. In particular, we focus on a relatively unexplored scenario, *i.e. incremental domain adaptation* for object detection assuming *weak-labeling*. Therefore, our algorithm is ready to improve existing source-oriented DPM-based detectors as soon as a little amount of labeled target-domain training data is available, and keeps improving as more of such data arrives in a continuous fashion. For achieving this, we follow a multiple instance learning (MIL) paradigm that operates in an incremental per-image basis. As proof of concept, we address the challenging scenario of adapting a DPM-based pedestrian detector trained with synthetic pedestrians to operate in real-world scenarios. The obtained results show that our incremental adaptive models obtain equally good accuracy results as the batch learned models, while being more flexible for handling continuously arriving target-domain data.

1 Introduction

Nowadays, classifiers play a core role in many computer vision tasks such as scene classification, object recognition, or object detection, among others. In many successful cases the classifier is learned from a training set containing both positive (examples) and negative samples (counter-examples). In this context, there are two relevant aspects worth to remind. First, collecting a training set is not a cost-free process since the required images must be acquired and the positive/negative samples labeled. In most of the cases, the labeling is

a tiresome manual operation prone to errors. Moreover, in many real applications image acquisition involves the deployment of equipment and personnel for days or months, *i.e.*, the images are not *just there*. Second, the underlying assumption for learning classifiers is that the training set and the deployment environment (testing) follow the same probability distribution regarding the features used by the classifiers.

In practice, there are different reasons that can break the constancy assumption of the probability distribution of the feature space. For instance, differences between training and testing data in terms of sensor quality, image resolution, predominant object poses and views, etc. Overall, these factors can cause a significant drop in the accuracy of the learned classifiers. Of course, if a sufficient amount of training data for the new testing *domain* is collected, we may consider retraining the classifiers. However, as we have pointed out, data collection can be costly and, therefore, in the general case doing so is not the optimal use of resources. Accordingly, reusing the existing classifiers by *adapting* them from the previous training environment (*source domain*) to the new testing one (*target domain*) is an approach worth to pursue and with increasing acceptance in the computer vision community [10, 11, 12, 16, 19, 20].

In this paper we focus on domain adaptation for object detection, using the *on-board pedestrian detection* task [8] as proof of concept. In particular, we will assume the use of the *deformable part-based model* (DPM) for representing the objects, since it is a recognized state-of-the-art method for both object detection in general [7] and pedestrian detection in particular [9]. Moreover, we focus on performing an *incremental domain adaptation* of DPM-based object detectors. The main benefit is to have an algorithm ready to improve existing source-oriented detectors as soon as a little amount of labeled target-domain training data is available, and keep improving as more of such data arrives in a continuous fashion. Note that, in opposition, a batch approach would wait until all the new labeled training data is available, and then run the adaptation by considering all the training samples at a time. In fact, we even consider a *weak-labeling* setting, which has two potential advantages. Firstly, gaining robustness to the variability on the bounding box location of each pedestrian, due to the fact of being a manually collected ground truth. Secondly, this allows to consider the scenario where the adapted detector self-labels target-domain training examples (*e.g.*, pedestrian bounding boxes) by detection; thus, reducing the manual work to the rejection of false positives (or acceptance of true ones).

Overall, as we will overview in section 2, we focus on a relatively unexplored scenario, *i.e.* incremental domain adaptation for object detection assuming weak-labeling. In section 3, we present our adaptation model as a weighted ensemble of source- and target-domain classifiers. This model is inspired in online transfer learning (OTL) [23]; however, contrarily to OTL, our method does not operate on a per-sample basis since it performs a per-image adaptation. In particular, the ensemble weights are time-dependent (*i.e.*, understanding time instants as those when new target-domain training data is available) to rely more on the source-domain classifier or on the target-domain one. At the same time, as explained in section 4, the target-domain classifier is also time-dependent and continuously updated from weakly-labeled target-domain training data. The weak labels are handled by following a multiple instance learning (MIL) paradigm for DPM training. In particular, we modify the CCCP (concave-convex procedure) at the core of DPM 5.0 [9] for operating in an incremental per-image basis.

We evaluate our overall proposal in section 5. In particular, following [20, 21], we address the challenging scenario of adapting a DPM-based pedestrian detector trained with synthetic pedestrians to operate in real-world scenarios. The obtained results show that our

incremental adaptive models obtain equally good accuracy results as the batch learned models, while being more flexible for handling continuously arriving target-domain data. Finally, in section 6 we summarize our main conclusions and future work.

2 Related Work

Domain adaptation is receiving growing attention in the computer vision community. However, most of the research in this line focus on object recognition [4, 16] and recently on object detection [12, 20]. The most related work to ours is the online transfer learning (OTL) framework [23], which is based on ensemble learning. It learns a classifier online with data from the target domain, and combines it with the source domain classifier. The combination weights are adjusted dynamically according to the loss of the two classifiers on the target domain samples. We extended this framework to incrementally adapt a detector. The recent work of [19] is also an extention of OTL while focusing on multiple category object recognition. The online domain adaptation of [12] has a similar setting to ours, while the employed Gaussian Process Regression requires a large number of samples from each testing image and thus only certain applications such as face detection can benefit from that method. The work of [11] proposes a continuous manifold domain adaptation method for an evolving visual domain, *e.g.*, traffic scene images taken at different time over the year. Different to [11], our method aims at performing a continuous domain adaptation with arbitrary target domain samples.

Our work is also related to multiple instance learning (MIL). MIL can be used for training with weakly-labeled data [2, 15] and several online MIL methods have been successfully applied to object tracking [1, 13, 14]. The batch mode adaptive multiple instance learning method [14] has a similar incremental learning strategy to ours, *i.e.*, using an online Passive Aggressive (PA) learning paradigm [18] for learning a new classifier based on a previously learned classifier. In contrast to previous MIL work, our learning algorithm is based on weak-label structural SVM (WL-SSVM) [9] and it can be applied to structural models, *e.g.*, DPM as we present in this paper. Our incremental MIL is a natural design for the OTL framework and we combine it with OTL for incremental domain adaptation.

3 Incremental Domain Adaptation Framework

We propose a frame by frame incremental domain adaptation framework for object detection. Suppose we are given a set of training samples $(\mathbf{x}_1, y_1, \mathbf{h}_1), \ldots, (\mathbf{x}_N, y_N, \mathbf{h}_N) \in \mathcal{X} \times \mathcal{Y} \times \mathcal{H}$, where $\mathcal{X}$ is the input space, $\mathcal{Y} = \{+1, -1\}$ is the label space, and $\mathcal{H}$ is the hypothesis or output space. The DPM [7], the decision function can be written as $f(\mathbf{x}) = \max_{\mathbf{h} \in \mathcal{H}} \mathbf{w}' \Phi(\mathbf{x}, \mathbf{h})$, where $\Phi(\mathbf{x}, \mathbf{h})$ is a joint feature vector.

Our method is an extension of the Online Transfer Learning (OTL) algorithm proposed in [23]. The basic idea is to learn an ensemble classifier $f^E(\mathbf{x})$ which is a weighted combination of the source domain classifier $f^S(\mathbf{x})$ and target domain classifier $f_t^T(\mathbf{x})$ at time t of the incremental learning task. We denote by γ_t^S and γ_t^T the combination coefficients. At time t, given a sample $\mathbf{x}$, the ensemble decision function is written as follows:

$$f^E(\mathbf{x}) = \gamma_t^S f^S(\mathbf{x}) + \gamma_t^T f_t^T(\mathbf{x}), \tag{1}$$

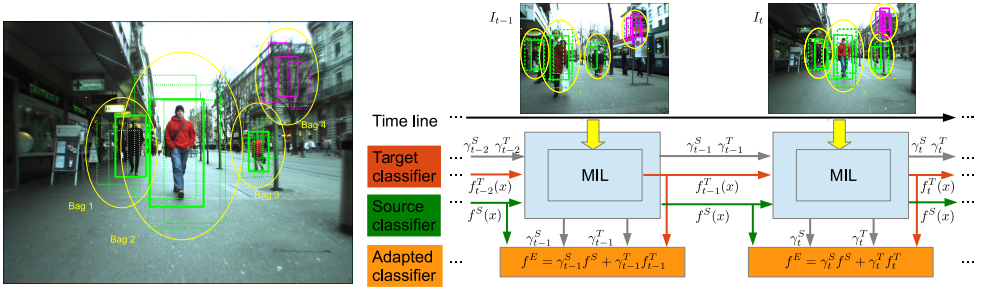

Figure 1: Incremental Multiple Instance Learning framework. (a) Multiple instance learning on a single frame. Bags are denoted by ellipses and their instances by rectangles. In this example, bags 1, 2, and 3 are positive ones, while bag 4 is negative. (b) Diagram of the frame by frame incremental adaptive learning. $f_t^T(\mathbf{x})$ is the classifier trained by multiple instance learning (MIL) with current target image, while the final target-domain adapted classifier is f^E.

where $f_t^T(\mathbf{x})$ is updated incrementally each time. Note that $f^S(\mathbf{x})$ and $f_t^T(\mathbf{x})$ are not independent as they maximize over the same $\mathbf{h}$ at training and testing time. In addition to updating $f_t^T(\mathbf{x})$, the two coefficients γ_t^S and γ_t^T are adjusted dynamically. The following updating scheme can be extended from OTL [23]:

$$\gamma_t^S = \frac{\gamma_{t-1}^S g_{t-1}(\bar{y}_i^S, y_i)}{\Gamma_{t-1}}, \quad \gamma_t^T = \frac{\gamma_{t-1}^T g_{t-1}(\bar{y}_i^T, y_i)}{\Gamma_{t-1}} \quad , \tag{2}$$

where $\Gamma_t = \gamma_t^S g_t(\bar{y}_i^S, y_i) + \gamma_t^T g_t(\bar{y}_i^T, y_i)$, $\bar{y}_i^S$ is the predicted label by f^S and $\bar{y}_i^T$ by f_{t-1}^T, $g_t(\bar{y}_i, y_i) = \frac{1}{N_t}\sum_{i=0}^{N_t} \exp\{-\frac{1}{2} l^*(\Pi(\bar{y}_i), \Pi(y_i))\}$, N_t is the number of target domain training samples at time t, $\Pi(s) = \max(0, \min(1, \frac{s+1}{2}))$ is a normalization function, and $l^*(\bar{y}, y) = (\bar{y} - y)^2$ is the square loss we use.

4 Learning in the target domain

In this section, we introduce the incremental learning of the classifier $f_t^T(\mathbf{x})$ in the target domain. We propose a method to train a DPM in a MIL manner using the weak-label Structural SVM (WL-SSVM). In Section 4.1 we first introduce the batch MIL learning with collected samples from a single frame and then in Section 4.2 we focus on the incremental setting, explaining the frame by frame incremental MIL learning of the DPM. Finally, we combine the incremental MIL with our proposed domain adaptation framework and present in detail the learning algorithm.

4.1 Multiple Instance Learning with WL-SSVM

The weak-label structural SVM (WL-SSVM) [2] is a discriminative training formalism for learning models from weakly-labeled samples. It generalizes the structural SVM, the latent

structural SVM and also includes the latent SVM as a special case. WL-SSVM minimizes the following objective function:

$$J(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N} \mathcal{L}_{surr}(\mathbf{w}, \mathbf{x}_i, y_i, \mathbf{h}_i), \tag{3}$$

where $C > 0$ is the trade-off parameter, $\mathcal{L}_{surr}$ is the surrogate training loss which is defined in terms of two different loss augmented predictions:

$$\begin{aligned}\mathcal{L}_{surr}(\mathbf{w}, \mathbf{x}_i, y_i, \mathbf{h}_i) = \quad & \max_{\mathbf{h}\in\mathcal{H}}[\mathbf{w}'\Phi(\mathbf{x}_i, \mathbf{h}) + \mathcal{L}_{margin}(y_i, \mathbf{h}_i, \mathbf{h})] - \\ & \max_{\mathbf{h}\in\mathcal{H}}[\mathbf{w}'\Phi(\mathbf{x}_i, \mathbf{h}) - \mathcal{L}_{output}(y_i, \mathbf{h}_i, \mathbf{h})].\end{aligned} \tag{4}$$

Let y be the predicted label according to the overlap of $\mathbf{h}_i$ and $\mathbf{h}$, *e.g.* using PASCAL criterion [8]. $\mathcal{L}_{margin}$ is the 0-1 loss, *i.e.*, if $y_i = y$, $\mathcal{L}_{margin} = 0$, otherwise 1. $\mathcal{L}_{output}$ is a 0-∞ loss, *i.e.*, if $y_i = y$, $\mathcal{L}_{output} = 0$, otherwise ∞. The overlap threshold in $\mathcal{L}_{margin}$ is 50% while in $\mathcal{L}_{output}$ is 70%. We refer the reader to [9] for more details.

We focus on training a DPM with samples from a single image and formulate the training as a multiple instance learning (MIL). Assume we received a training image I_t at time t. We first extract all the positive samples from I_t. These positive samples can be either obtained from human annotations, a pre-trained detector, or even a pre-trained detector aided by a human oracle. These annotations are generally in a form of bounding boxes, which may be prone to errors and not necessarily the best ground truth for a supervised learning algorithm. Therefore, they can be regarded as weakly-labeled samples. Image windows with low overlapping with the positive samples are considered negative samples.

We consider the task of training a DPM detector with weakly-labeled samples in a MIL manner. We treat each sample, $(\mathbf{x}_i, y_i, \mathbf{h}_i)$, as a *initial ground truth* and additionally we include multiple possible outputs related to this sample to form a bag of instances. We denote by $\mathcal{B}^+$ a positive bag and $\mathcal{B}^-$ a negative bag. The bags and instances are illustrated in Figure 1 (a). The alternative outputs $\mathbf{h}$ in $\mathcal{B}^+$ are collected according to the following condition:

$$\mathbf{w}'\Phi(\mathbf{x}_i, \mathbf{h}_i) - \mathbf{w}'\Phi(\mathbf{x}_i, \mathbf{h}) < \mathcal{L}_{margin}(y_i, \mathbf{h}_i, \mathbf{h}) + \varepsilon \quad \textit{and} \quad \mathbf{h} \neq \mathbf{h}_i, \tag{5}$$

where ε is a constant tolerance parameter set in practice to 0.001. It is equivalent to find the most violate outputs of sample $\mathbf{x}_i$. The negative bags are collected from the image patches not contained in $\mathcal{B}^+$. We restrict each positive bag to contain only one *belief* instance, *i.e.*, $\mathbf{h}^*$, and this instance will play the role of *ground truth* output. For the negative bag $\mathcal{B}^-$, there is a constant background *belief* instance $\mathbf{h}^*$, which corresponds to the feature vector $\mathbf{0}$. We iteratively train a DPM model as follows: (1) Fixing the *belief* instances in each bag and learning a new model $\mathbf{w}$ with Eq. (3). (2) Updating the positive *belief* instances in each bag with the current model $\mathbf{w}$. The learning terminates when the *belief* instances in each positive bag do not change. The *belief* in $\mathcal{B}^+$ is computed by the following equation:

$$\mathbf{h}^* = arg\max_{\mathbf{h}\in\mathcal{H}}[\mathbf{w}'\Phi(\mathbf{x}_i, \mathbf{h}) - \mathcal{L}_{output}(y_i, \mathbf{h}_i, \mathbf{h})]. \tag{6}$$

The complete algorithm of WL-SSVM-based MIL is illustrated in Alg. 1.

4.2 Incremental Multiple Instance Learning

Algorithm 1 Multiple Instance Learning with WL-SSVM

Input: target-domain training image I_t, initial model $\mathbf{w}_0$
Output: new model $\mathbf{w}$
0: $\mathbf{w} \leftarrow \mathbf{w}_0$
1: Collect positive samples from I_t using root bounding boxes $\{(\mathbf{x}_i, y_i, \mathbf{h}_i)\}$.
Part locations in $\mathbf{h}_i$ are initialized by finding the best detection using $\mathbf{w}_0$.
2: Initialize positive bags $\mathcal{B}_i^+ \leftarrow \{(\mathbf{x}_i, +1, \mathbf{h}_i)\}$.
3: Augment positive bags by Eq. (5), $\mathcal{B}_i^+ \leftarrow \{(\mathbf{x}_i, +1, \mathbf{h}) | \mathbf{h} \in \mathcal{H}\}$,
initialize the *belief* instance of $\mathcal{B}_i^+$ by $\mathbf{h}^* \leftarrow \mathbf{h}_i$.
4: Collect negative samples and build one negative bag $\mathcal{B}^- \leftarrow \{(\mathbf{x}_j, -1, \mathbf{h}_j)\}$,
initialize the *belief* instance of $\mathcal{B}^-$ with $\mathbf{0}$.
5: **repeat**
6: Update $\mathbf{w}$ by minimizing the objective function (3).
7: Compute the new *belief* instances of each bag by (6).
8: **until** *belief* instances do not change

Algorithm 2 Incremental Domain Adaptation

Input:
source classifier f^S
target images $\{I_t, t \in [1, N]\}$
Output: $f^E = \gamma_N^S f^S + \gamma_N^T f_N^T$
0: $f_0^T \leftarrow f^S$, $\gamma_1^S = \gamma_1^T \leftarrow 0.5$
1: **for** t=1,2, ..., N, **do**
2: Receive image I_t, collect samples $\mathcal{D} \leftarrow \{(\mathbf{x}_i, y_i)\}$.
3: Predict $\bar{y}_j^S$ by f^S, and $\bar{y}_j^T$ by f_{t-1}^T, $j \in \{1, N_t\}$.
4: Compute γ_t^S and γ_t^T by (2).
5: Generate training bags for MIL (Alg. 1 line 2-4).
6: Learn f_t^T with the collected bags (Alg. 1 line 5-8).
7: **end for**

We apply an incremental learning strategy similar to [14] for training a frame-by-frame adaptive classifier. Assume we receive an image I_t at time t and we learn f_t^T on that image by updating f_{t-1}^T learned at time $t-1$. Motivated by the online learning algorithms [18] and [14], we define f_t^T on instance $\mathbf{x}$ as follows:

$$f_t^T(\mathbf{x}) = \max_{\mathbf{h} \in \mathcal{H}} [\mathbf{w}'_{t-1} \Phi(\mathbf{x}, \mathbf{h}) + (\mathbf{w}'_t - \mathbf{w}'_{t-1}) \Phi(\mathbf{x}, \mathbf{h})] = f_{t-1}^T(\mathbf{x}) + \Delta f_t^T(\mathbf{x}), \tag{7}$$

where $\Delta f_t^T(\mathbf{x})$ is the perturbation function. Given the training bags with instances $\mathbf{x}_i, \ldots, \mathbf{x}_{N_t}$, we learn the parameters $\mathbf{w}_t$ in the perturbation function by minimizing the following objective function:

$$J(\mathbf{w}_t) = \frac{1}{2} \|\mathbf{w}_t - \mathbf{w}_{t-1}\|^2 + C \sum_{i=1}^{N_t} \mathcal{L}_{surr}(\mathbf{w}_t, \mathbf{x}_i, y_i, \mathbf{h}_i). \tag{8}$$

The optimization of the objective function (8) can be solved by L-BFGS [17] in the primal form. It only requires computing the objective value and the partial derivatives with respect to $\mathbf{w}_t$. With the above learning strategy, f_t^T can be embedded into the proposed domain adaptation framework in Section 3. The complete algorithm is presented in Alg. 2.

5 Experiments

We evaluate the proposed method on several pedestrian datasets. We use a synthetic dataset [22] to train our source domain DPM detector, and adapt it to multiple real-world datasets. We use the three subsets from the ETH dataset [8] as our target domains, namely ETH0, ETH1 and ETH2, as they are recorded in three different scenarios and under different illumination conditions.We follow the standard Caltech evaluation criterion [3] and plot the average miss rate *vs* false positive per image (FPPI) curves. Each target dataset is randomly split into a train/test pair. In particular, in each case, we use 50 random images for training and the rest for testing. We run each experiment three times and compute the resulting mean and standard deviation per experiment.

5.1 Supervised Domain Adaptation

We train different types of classifiers as shown in Table 1. First, we compare our incremental domain adaptive classifier to the batch-mode learned classifiers as it can be seen in the first column of Figure 2.

Table 1: Different types of learned classifiers.

SRC	Source domain classifier (no adaptation).
BAT-TAR	Target domain classifier trained in batch-mode with the selected labeled 50 images. It is simply a re-training of DPM, *i.e.*, no adaptation.
INC-Dyn-[W1=Ws]	Incremental domain adaptation where the coefficients of the source and target classifiers are updated during the training, and the target model $\mathbf{w}$ is initialized by $\mathbf{w}^S$.
INC-Fix-[W1=0]	Incremental domain adaptation where the coefficients of the source and target classifiers are fixed, and the target model $\mathbf{w}$ is initialized with vector $\mathbf{0}$.
INC-Fix-[W1=Ws]	Incremental domain adaptation where the coefficients of the source and target classifiers are fixed, and the target model $\mathbf{w}$ is initialized with $\mathbf{w}^S$.
BAT-ADP	Batch domain adaptation by considering all the target training samples at once. We extend Eq. (8) to train with samples from multiple images, *i.e.* $\min \frac{1}{2}\|\mathbf{w}^T - \mathbf{w}^S\|^2 + C\sum_{i=1}^{N} \mathcal{L}_{surr}(\mathbf{w}^T, \mathbf{x}_i, y_i, \mathbf{h}_i)$.

From the experimental results, we can see that both BAT-ADP and INC-Dyn-[W1=Ws] are effectively adapted to the target domain, comparing to the source classifier SRC. INC-Dyn-[W1=Ws] accuracy is not too far from the one of BAT-ADP in ETH1 and ETH2 datasets. Batch learned BAT-TAR turns out to have very low accuracy results due to the low number of target-domain training data, showing the convenience of using such data for performing domain adaptation. Next, we investigate the impact of different parameters in the learning of the incremental domain adaptation method as can be appreciated in the second column of Figure 2.

With the coefficients fixed, INC-Fix-[W1=Ws] outperforms INC-Fix-[W1=0] significantly on the three datasets, showing the importance of the initialization of the incremental target domain classifier $f_t^T(\mathbf{x})$. With dynamic updating, INC-Dyn-[W1=Ws] achieved better accuracy than INC-Fix-[W1=Ws] on all the target datasets, showing the effectiveness of the incremental adaptive learning.

To further understand the dynamic changes during the incremental adaptive learning, we plot the average square loss (*i.e.*, $l^*(\bar{y}, y) = (\bar{y} - y)^2$) and coefficients (*i.e.*, γ_t^S, γ_t^T) of the source $f^S(\mathbf{x})$ and target $f_t^T(\mathbf{x})$ classifiers in each training image (or incremental iteration). Figure 3 shows the results on the three training datasets. The incrementally learned target domain classifier $f_t^T(\mathbf{x})$ shows constantly lower average loss on the training images. The

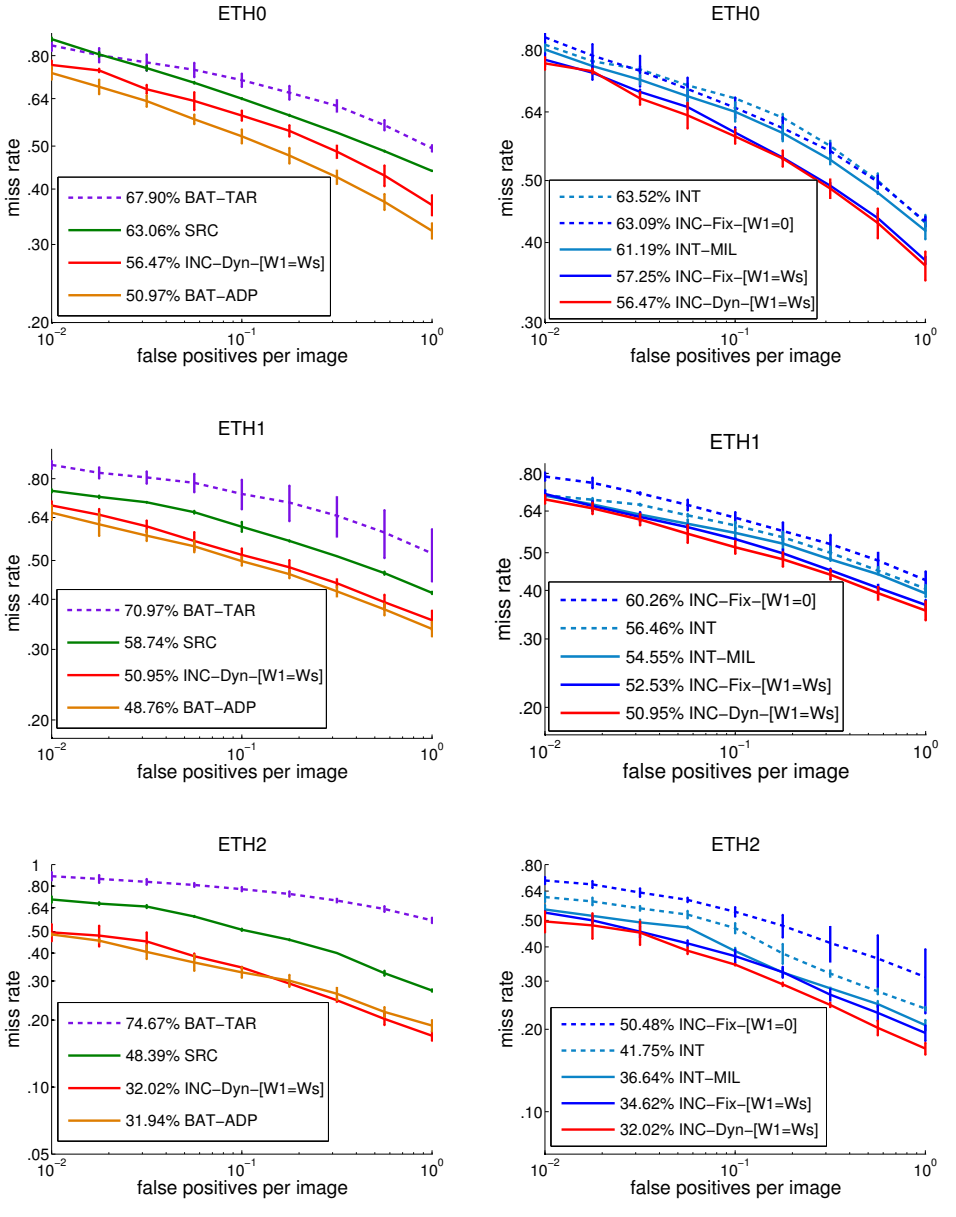

Figure 2: Results of adapting a DPM pedestrian detector trained with synthetic images to operate in ETH pedestrian dataset.

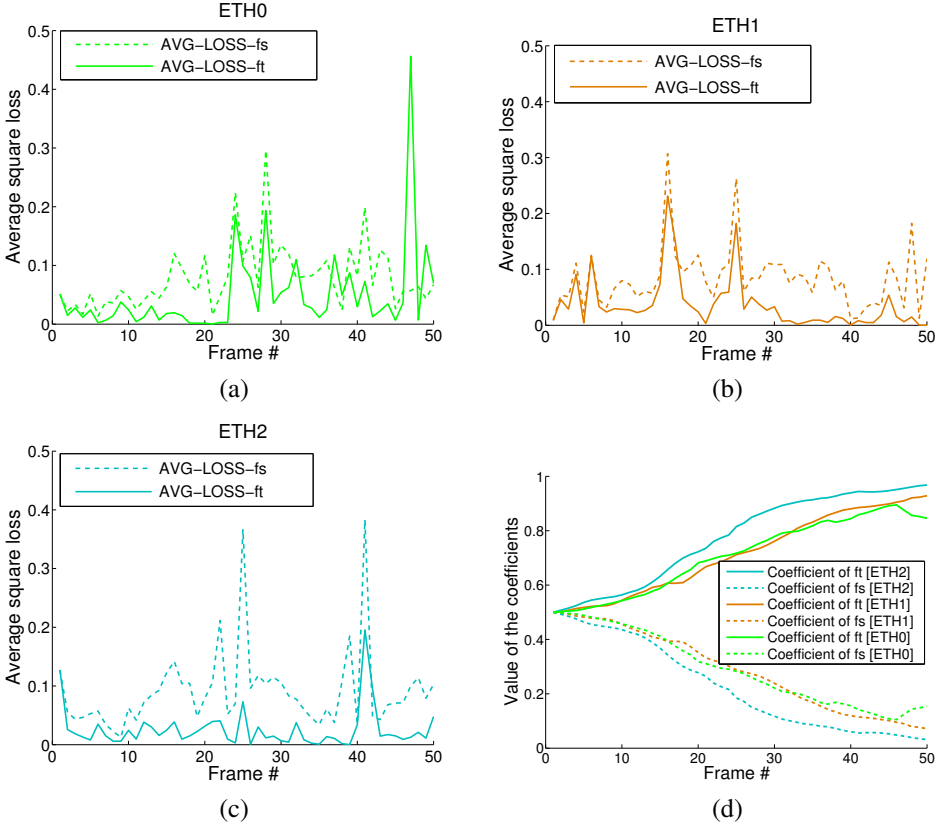

Figure 3: (a)-(c) Average square loss of the source and target classifier in each iteration. (d) Coefficient (γ_t^S, γ_t^T) changes at each iteration.

curves of γ_t^S and γ_t^T show that the source classifier plays a less important role as $f_t^T(\mathbf{x})$ gradually adapts to the target domain. Figure 3 (d) also shows that the source classifier is adapted to the target domain rapidly, as γ_t^T reached more than 0.8 after the first 45 frames.

5.2 Incremental Domain Adaptation with Human in the Loop

In this experiment, we use unlabeled target domain samples for the incremental adaptive learning. We designed an interactive incremental learning framework to include a human oracle in the loop. The human oracle is allowed to only remove the false positives by simple clicks in each training image. In this case, the training samples collected by the source detector are largely weakly-labeled. We compare the adapted model trained with MIL, denoted by INT-MIL, with a model that does not include MIL, denoted by INT (INT stands for interactive).

The results are shown in the second column of Figure 2. It can be seen that INT-MIL improves the accuracy of the adapted classifier around 2 percentage points compare to INT, showing the effectiveness of the MIL. However, there is still a gap between INT-MIL and the fully supervised learning with original ground truth, *i.e.*, INC-Dyn-[W1=Ws]. This may be due to the false negative samples, *i.e.*, some pedestrians are not considered during training.

6 Conclusion

In this paper, we present an incremental domain adaptation framework applied to the deformable part-based model for object detection. A dynamic adaptation strategy, inspired in an online transfer learning method, learns an ensemble of the source and target domain classifiers. At each iteration, a new target domain classifier is learned on the underlying image. We apply weak-label structural SVM for handling weakly-labeled samples and formulate the training in a multiple instance learning paradigm. The conducted experiments on pedestrian detection show the effectiveness of the proposed method. The incremental domain adaptation achieves comparable accuracy results to the batch learned model while being more flexible for learning with continuously coming target domain data. In the future, we plan to focus on improving the incremental domain adaptation with unlabeled target domain images.

Acknowledgment

This work is supported by the Spanish MICINN projects TRA2011-29454-C03-01 and TIN2011-29494-C03-02, the Chinese Scholarship Council (CSC) grant No.2011611023 and Sebastian Ramos' FPI Grant BES-2012-058280.

References

[1] B. Babenko, M. Yang, and S. Belongie. Robust object tracking with online multiple instance learning. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 33(8): 1619–1632, 2011.

[2] M. Blaschko, A. Vedaldi, and A. Zisserman. Simultaneous object detection and ranking with weak supervision. In *Advances in Neural Information Processing Systems*, 2010.

[3] P. Dollár, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: an evaluation of the state of the art. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 34(4): 743–761, 2012.

[4] L. Duan, D. Xu, and IW. Tsang. Learning with augmented features for heterogeneous domain adaptation. In *Int. Conf. on Machine Learning*, 2012.

[5] A. Ess, B. Leibe, and L. Van Gool. Depth and appearance for mobile scene analysis. In *Int. Conf. on Computer Vision*, 2007.

[6] M. Everingham, L. Van Gool, C.K.I. Williams, J. Winn, and A. Zisserman. The PASCAL visual object classes (VOC) challenge. *Int. Journal on Computer Vision*, 88(2): 303–338, 2010.

[7] P. Felzenszwalb, R.B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010.

[8] D. Gerónimo, A.M. López, A.D. Sappa, and T. Graf. Survey of pedestrian detection for advanced driver assistance systems. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 32(7):1239–1258, 2010.

[9] R.B. Girshick. *From Rigid Templates to Grammars: Object Detection with Structured Models*. PhD thesis, The University of Chicago, Chicago, IL, USA, 2012.

[10] J. Hoffman, B. Kulis, T. Darrell, and K. Saenko. Discovering latent domains for multisource domain adaptation. In *European Conf. on Computer Vision*, 2012.

[11] J. Hoffman, T. Darrell, and K. Saenko. Continuous manifold based adaptation for evolving visual domains. In *IEEE Conf. on Computer Vision and Pattern Recognition*, 2014.

[12] V. Jain and E. Learned-Miller. Online domain adaptation of a pre-trained cascade of classifiers. In *IEEE Conf. on Computer Vision and Pattern Recognition*, 2011.

[13] M. Li, J. Kwok, and B. Lu. Online multiple instance learning with no regret. In *IEEE Conf. on Computer Vision and Pattern Recognition*, 2010.

[14] W. Li, L. Duan, IW. Tsang, and D. Xu. Batch mode adaptive multiple instance learning for computer vision tasks. In *IEEE Conf. on Computer Vision and Pattern Recognition*, 2012.

[15] Z. Lin, G. Hua, and L. Davis. Multiple instance feature for robust part-based object detection. In *IEEE Conf. on Computer Vision and Pattern Recognition*, 2009.

[16] K. Saenko, B. Hulis, M. Fritz, and T. Darrel. Adapting visual category models to new domains. In *European Conf. on Computer Vision*, 2010.

[17] M. Schmidt. Minconf - projection methods for optimization with simple constraints in matlab. http://www.di.ens.fr/ mschmidt/Software/minConf.html.

[18] S. Shalev-shwartz, K. Crammer, O. Dekel, and Y. Singer. Online passive-aggressive algorithms. In *Advances in Neural Information Processing Systems*, 2003.

[19] T. Tommasi, F. Orabona, M. Kaboli, B. Caputo, and C. Martigny. Leveraging over prior knowledge for online learning of visual categories. In *British Machine Vision Conference*, 2012.

[20] D. Vázquez, A.M. López, J. Marín, D. Ponsa, and D. Gerónimo. Virtual and real world adaptation for pedestrian detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 36(4):797–809, 2014.

[21] J. Xu, S. Ramos, D. Vázquez, and A.M. López. Domain adaptation of deformable part-based models. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, in press, 2014.

[22] J. Xu, D. Vazquez, A.M. Lopez, J. Marin, and D. Ponsa. Learning a part-based pedestrian detector in a virtual world. *IEEE Trans. on Intelligent Transportation Systems*, in press, 2014.

[23] P. Zhao and S. Hoi. OTL: A framework of online transfer learning. In *Int. Conf. on Machine Learning*, 2010.