Hierarchical online domain adaptation of deformable part-based models

Jiaolong Xu[1], David Vázquez[2], Krystian Mikolajczyk[3] and Antonio M. López[1]

***Abstract*— We propose an online domain adaptation method for the deformable part-based model (DPM). The online domain adaptation is based on a two-level hierarchical adaptation tree, which consists of instance detectors in the leaf nodes and a category detector at the root node. Moreover, combined with a multiple object tracking procedure (MOT), our proposal neither requires target-domain annotated data nor revisiting the source-domain data for performing the source-to-target domain adaptation of the DPM. From a practical point of view this means that, given a source-domain DPM and new video for training on a new domain without object annotations, our procedure outputs a new DPM adapted to the domain represented by the video. As proof-of-concept we apply our proposal to the challenging task of pedestrian detection. In this case, each instance detector is an exemplar classifier trained online with only one pedestrian per frame. The pedestrian instances are collected by MOT and the hierarchical model is constructed dynamically according to the pedestrian trajectories. Our experimental results show that the adapted detector achieves the accuracy of recent supervised domain adaptation methods (*i.e.*, requiring manually annotated target-domain data), and improves the source detector more than 10 percentage points.**

I. Introduction

Classifiers play a core role in many computer vision tasks. Training an accurate classifier usually requires a large amount of labeled training data. 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. However, a possible scenario is that the deployment environment (testing domain) does not follow the same probability distribution as the training domain (*i.e.* the dataset bias [1]), or the testing data may be collected dynamically and its distribution can vary over time. For example, an on-board pedestrian detection system may face large variety of scenarios during the driving, *e.g.*, different cities, different weathers, different seasons, *etc.* All these can cause a significant drop in the accuracy of the learned classifiers.

One may consider updating the classifiers by adding training data for the new target domain. However, required data collection may be not practical and limits the range of possible applications. Accordingly, reusing the existing classifiers by *adapting* them from one training environment (*source domain*) to the new testing one (*target domain*) is an approach increasingly exploited in the computer vision community [2], [3], [4], [5], [6], [7]. In this work, we focus on the *domain adaptation* of DPM-based object detectors due to the high performance usually exhibited by such rich object representations.

Domain adaptation of DPMs has been explored in [8], which demonstrates that a DPM detector trained with synthetic data can be adapted to various real-world datasets with a relatively few labeled real-world (target domain) data. However, the proposed domain adaptation methods have several limitations. First, they require manual annotation in the target domain (*i.e.*, *supervised domain adaptation*); Second, all newly annotated training data needs to be available for the adaptation, *i.e.* following a typical batch learning mode. To address these problems, in this work, we explore an *online domain adaptation* technique for the DPM-based object detectors. In particular, given an unlabeled training video of a new domain, our method automatically adapts the current DPM without human intervention (*i.e.*, *unsupervised domain adaptation*) and without the data used to learn it. This can be a continuous process as new videos arrive.

The main benefit of an *unsupervised online domain adaptation* is to improve existing source-oriented detectors as soon as a new *unlabeled* target-domain training data is available, and keep improving as more of such data arrives in a continuous fashion. We build on the recent proposal of hierarchical adaptive structured SVM (HA-SSVM) [9], originally defined as a batch-mode supervised domain adaptation procedure. Our online domain adaptation is based on a two-level hierarchical adaptation tree, which consists of instance detectors in the leaf nodes and a category detector at the root node. Each instance detector is an exemplar classifier which is trained online with only one object per frame. The object instances are collected by multiple object tracking (MOT) and the hierarchical model is constructed dynamically according to the object trajectories. As proof-of-concept we apply our proposal to the challenging task of pedestrian detection. Fig. 1 illustrates the main idea.

The rest of the paper is organized as follows. In Section II, we review the related work on domain adaptation. Section III elaborates the proposed method. We first introduce the overview of the online domain adaptation framework. Then we go to the details of each step, including the MOT process and the learning algorithm. Section IV shows the experimental results on ETH datasets. Finally, Section V draws the main conclusions.

[1]Jiaolong Xu and Antonio M. López are with the Computer Vision Center (CVC) and the Computer Science Department at the Universitat Autònoma de Barcelona. jiaolong, antonio@cvc.uab.es

[2]David Vázquez is with the CVC. dvazquez@cvc.uab.es

[3]Krystian Mikolajczyk is with the Department of Electrical and Electronic Engineering at the Imperial College London. k.mikolajczyk@imperial.ac.uk

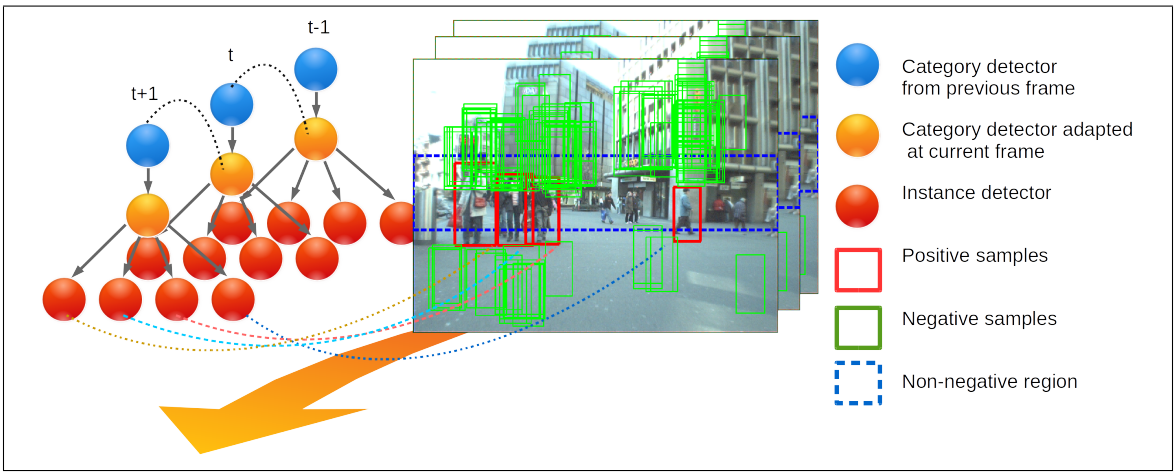

Fig. 1. Our hierarchical unsupervised online domain adaptation framework for DPMs.

II. Related work

Our work is related to domain adaptation, which is an emergent topic in computer vision in recent years. Most of the work focus on object recognition [2], [10] and recently on object detection [3], [7], [8]. The domain adaptation methods in computer vision can be categorized into two main groups, *i.e.* model-transform-based and feature-transform-based. Other methods also consider a joint learning of mode transformation and feature transformation, *e.g.*, [10], [11].

We refer to [12] for a comprehensive review of domain adaptation techniques. We mainly review recent proposals in the context of object detection. In [8], the adaptive SVM (A-SVM) is extended to structured SVM (A-SSVM) for training domain adaptive DPM. The A-SSVM is further developed in [9] to build a hierarchical model for progressive domain adaptation. Other works also proposed domain adaptation for general object detection, *e.g.*, [13], [14]. In this work, we extend the hierarchical domain adaptation framework of [9] and tackle the more challenging unsupervised online setting.

Due to the challenging setting, online domain adaptation is a relatively little explored scenario. The online transfer learning (OTL) framework [15], which is based on ensemble learning, has been proven a successful method. 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. The later work of [4] extended OTL and addressed the problem of multiple category object recognition. Also inspired by OLT, recently, an incremental domain adaptation method for DPM was proposed in [16]. The incremental domain adaptation method is based on multiple instance learning (MIL), which shares some properties of online MIL, *e.g.* object tracking [17], [18], [19]. Although [16] can also perform domain adaptation with incrementally collected testing data, it requires labeled target domain data. Even at the semi-supervised setting, a human oracle is required to click out false positives. A recent work of [20] proposed a method for classifier adaptation at prediction time, which is orthogonal to ours. They target distribution changes due to changing class proportions while our aim at adapting to changes in appearance of the classes themselves.

Similar to our hierarchical model, a recent work of [21] proposed a category-to-instance detector for improving tracking. Target objects are identified with a pre-trained category detector and object identity across frames is established by individual-specific detectors. The individual detectors are re-trained online from a single positive example whenever there is a coincident category detection. The method is built on a boosting classifier framework while our method directly models the category-to-instance adaptation in a unified HA-SSVM optimization framework. The final goal of [21] is online tracking, while ours is to obtain a generic target domain adapted detector.

III. Hierarchical online domain adaptation (HOLDA)

We first introduce the overall framework of the hierarchical model. Then we elaborate how to incorporate multiple object tracking into the pipeline for generating trajectories. Later, we formulate the learning as hierarchical adaptive SSVM and finally, we detail the algorithm.

A. Model

The hierarchical online adaptation framework is shown in Fig. 1. Given a target domain sequence, we first apply the source domain detector to collect the detected bounding boxes (red boxes). Then, MOT is used to generate trajectories and also remove some false positives. After generating the trajectories, our hierarchical model can be learned frame by frame using online adaptive SSVM. At each frame, the hierarchical model consists of instance detectors at the leaf nodes (red balls in Fig. 1) and a category detector at the second layer (the orange ball). The adaptation is executed in a progressive manner, *i.e.*, category detector (orange ball at time t), is adapted from the previously adapted category detector (blue ball at time t, adapted from orange ball at time $t-1$). At $t=0$, the category detector is initialized by the source domain detector. The instance detectors are

adapted concurrently with the category detector of the current frame. The hierarchical model is constructed dynamically according to the trajectories at current frame, *i.e.*, each instance detector corresponds to one trajectory. Each instance detector is essentially an exemplar classifier which is trained using only one positive example (red bounding box) and many negative examples (green bounding boxes). The negative examples are collected from the same frame, but outside the *non-negative* area. The non-negative region (blue dash rectangle) is defined using prior geometry knowledge, which can avoid accidentally introducing false negatives as background examples.

B. Generating trajectories by MOT

We use MOT to provide trajectory information for the hierarchical online learning. The MOT requires bounding boxes from the source domain detector as input and outputs the optimized trajectories which may recover some missing detections and eliminate false detections from the source detections. Besides the selected detections, the trajectories are directly used to build the hierarchical online model, *i.e.*, each trajectory corresponds to a leaf node in the adaptation tree.

In this work, we implemented a simple motion-based multiple object tracking algorithm based on Kalman filter. Though more sophisticated state-of-the-art MOT algorithms can be readily incorporated in our system, *e.g.* [22], we found our simple MOT has already given promising results. Our MOT method can be divided into three parts: (1) detecting pedestrians at each frame (this is provided by the source detector), (2) associating the detections corresponding to the same pedestrian over time, (3) evaluating each trajectory to identify the reliable ones. The method is summarized in Alg. 1. The reliability of a trajectory is estimated according to the length and the *confidence*. The length of a trajectory is is defined as the number of frames being active. The confidence of a trajectory is measured by averaging the detection scores of the associated bounding boxes. If the length or confidence is lower than a predefined threshold, the trajectory is defined as not reliable, otherwise as reliable.

Algorithm 1 Motion-based Multiple Object Tracking

Input: N: length of the sequence
$D_s = \{d_s(t)|t \in [1, N]\}$: set of bounding boxes detected by the source domain, where $d_s(t)$ are bounding boxes at frame t.
Output: $T^*(t)$:set of reliable tracks centred at frame t
0: Initialize the tracks $T^*(1)$ by $d_s(1)$.
1: **for** t=2, ..., N, **do**
2: Detect $d_s(t)$.
3: Predict new locations of active tracks in $T^*(t-1)$ by Kalman filter.
4: Data association: assign detections $d_s(t)$ to the active tracks in $T^*(t-1)$ using [23].
5: Update tracks:
  5.1: Correct the location estimate of Kalman filter for each continued track.
  5.2: Delete lost tracks.
  5.3: Create new tracks from unassigned detections.
6: Evaluate the reliability of the tracks and remove unreliable tracks in $T^*(t)$.
7: **end for**

C. Learning with HA-SSVM

We first review the training of DPM and then introduce the formulation of HA-SSVM proposed in [9]. Later, we elaborate how to extend HA-SSVM to online domain adaptation. We denote our method by *HOLDA*.

As in [24], we formulate the learning of DPM as a latent structured SVM. 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, *i.e.*, the configuration of object parts. We denote the features as joint feature vectors $\Phi(\mathbf{x}, \mathbf{h})$. In the DPM case [25], $\mathbf{h}$ is not given and is therefore treated as a latent variable during training. DPM training aims to learn an optimum $\mathbf{w}$ which encodes the appearance parameters and deformation coefficients. The objective function can be defined as follows:

$$\begin{underbrace{\min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N} \max_{\widehat{y},\widehat{\mathbf{h}}} [\mathbf{w}'\Phi(\mathbf{x}_i, \widehat{\mathbf{h}}) + L(y_i, \widehat{y}, \widehat{\mathbf{h}})]}_{convex} \\ \underbrace{-C\sum_{i=1}^{N} \max_{\mathbf{h}} \mathbf{w}'\Phi(\mathbf{x}_i, \mathbf{h}),}_{concave}} \quad (1)$$

where parameter C is the relative penalty scalar parameter, $L(y_i, \widehat{y}, \widehat{\mathbf{h}})$ represents the loss function, $\widehat{y}$ the predicted label, and y_i the ground truth label. In particular, we use 0-1 loss for object detection, *i.e.*, $L(y_i, \widehat{y}, \widehat{\mathbf{h}}) = 0$ if $\widehat{y} = y_i$ and 1 otherwise. The objective function (1) can be viewed as minimizing the sum of a convex and concave function and it can be solved by the general Convex-Concave Procedure (CCCP), which is a iterative procedure that guarantees the convergence to a local minimum or a stationary point of the objective function.

The HA-SSVM is extended from the adaptive SSVM (A-SSVM) [8], which is a model-transform-based domain adaptation method. Given the source model $\mathbf{w}^S$, the basic idea of A-SSVM is to learn a new decision boundary in the target domain close to the original source decision one. We denote by $\mathbf{w}$ the target domain model, $\Delta\mathbf{w} = \mathbf{w} - \mathbf{w}^S$, A-SSVM solve the following optimization problem: $\min_{\Delta\mathbf{w}} \mathcal{R}(\Delta\mathbf{w}) + C\mathcal{L}(\Delta\mathbf{w}, \mathcal{D}_l^T)$, where $\mathcal{R}$ is a regularizer, $\mathcal{L}$ represents the loss term on target data, and C is a penalty scalar parameter as in (1). Assume multiple target domain exists, instead of doing isolated single source-to-target adaptation with A-SSVM, HA-SSVM organizes target domains into a hierarchical adaptation tree. A-SSVM is served as a basic element for each parent-to-child adaptation. We refer

the readers to [9] for more details. In the following, we give concrete example of applying such hierarchical adaptation strategy for online domain adaptation.

Assume we have target domain images I_t, $i \in [1, N]$. Without losing generality, assume at frame t, we have 3 pedestrians. The category model we have learned at frame $t-1$ is denoted by $\mathbf{w}_c^{t-1}$. The category model and instance models at frame t are denoted by $\mathbf{w}_c^t$, $\mathbf{w}_{i_j}^t, j \in [1, 3]$ respectively. $\mathbf{w}_{i_j}^t$ is the parameter of the instance classifier i and it is learned with pedestrian example j and all the negative examples in frame t. We denote the training examples for $\mathbf{w}_{i_j}^t$ by $\mathcal{D}^{T_j}$. Then the objective function of HOLDA is written as follows:

$$\begin{aligned} J(\mathbf{w}) &= \frac{1}{2}\|\mathbf{w}_c^t - \mathbf{w}_c^{t-1}\|^2 + C\sum_{j=1}^{3}\mathcal{L}(\mathbf{w}_c^t; \mathcal{D}^{T_j}) \\ &+ \frac{1}{2}\|\mathbf{w}_{i_1}^t - \mathbf{w}_c^t\|^2 + C\mathcal{L}(\mathbf{w}_{i_1}^t; \mathcal{D}^{T_1}) \\ &+ \frac{1}{2}\|\mathbf{w}_{i_2}^t - \mathbf{w}_c^t\|^2 + C\mathcal{L}(\mathbf{w}_{i_2}^t; \mathcal{D}^{T_2}) \\ &+ \frac{1}{2}\|\mathbf{w}_{i_3}^t - \mathbf{w}_c^t\|^2 + C\mathcal{L}(\mathbf{w}_{i_3}^t; \mathcal{D}^{T_3}) \end{aligned} \tag{2}$$

where $t \in [0, N]$, $\mathbf{w}_c^0 = \mathbf{w}^S$, $C > 0$ is the trade-off parameter. For DPM, $\mathcal{L}(\mathbf{w}; \mathcal{D})$ is the training loss which is defined as: $\mathcal{L}(\mathbf{w}; \mathcal{D}) = \sum_{i=1}^{N} \max_{\widehat{y},\widehat{\mathbf{h}}}[\mathbf{w}'\Phi(\mathbf{x}_i, \widehat{\mathbf{h}}) + L(y_i, \widehat{y}, \widehat{\mathbf{h}})] - \sum_{i=1}^{N} \max_{\mathbf{h}} \mathbf{w}'\Phi(\mathbf{x}_i, \mathbf{h})$. where $L(y_i, \widehat{y}, \widehat{\mathbf{h}})$ is the 0-1 loss as in 1.

Equation 2 follows a multi-task learning paradigm form, where the optimization of each $\mathbf{w}_{i_j}^t$ can be understood as an individual task. After training with N frames, we obtained an adapted classifier with parameter $\mathbf{w}_c^N$. At testing time, we can directly apply the linear decision function:

$$f(\mathbf{x}) = \max_{\mathbf{h}\in\mathcal{H}} \mathbf{w}_c^{N\prime}\Phi(\mathbf{x}, \mathbf{h}) \ . \tag{3}$$

Connecting with our MOT, the overall algorithm of hierarchical online domain adaptation is described in Alg. 2. We denote by HOLDA-MOT the proposed method. Given a source domain trained detector $\mathbf{w}^S$ and target domain sequence of N frames, the first step is to apply MOT proposed in Alg. 1 to obtain refined trajectories $T^*(t), t \in [1, N]$. With these trajectories, we also obtain refined detections $d_s^*(t)$ on each frame. $d_s^*(t)$ consists of the associated bounding boxes of the trajectories, which are more confident detections than the original ones, *i.e.*, $d_s(t)$. At frame t, we can build the hierarchical model according to the current trajectory $T^*(t)$, *i.e.* the leaf nodes are corresponding to the individual pedestrians $d_s^*(t)$ and they will be used to train instance detectors $\mathbf{w}_i^t$. Then we extract negative samples from the background of frame t. A *non-negative* region is used to avoid selecting false negative examples. With the positive and negatives examples, we can optimize the hierarchical model at current frame and obtain an adapted category detector $\mathbf{w}_c^t$. In the next frame $t+1$, $\mathbf{w}_c^t$ will be used as *source* detector and adapted to $\mathbf{w}_c^{t+1}$. The intuition behind this is that at frame t the knowledge from the previous frames are encoded in $\mathbf{w}_c^t$; then, by adapting $\mathbf{w}_c^t$ to $\mathbf{w}_c^{t+1}$, $\mathbf{w}_c^{t+1}$ keeps the knowledge of the previous frames at the same time that learns from the examples at frame $t+1$. In this way, the final adapted target domain detector is $\mathbf{w}_c^N$.

Algorithm 2 HOLDA-MOT

Input:
Target domain sequence $I_t, t \in [1, N]$
Source domain detector $\mathbf{w}^S$
Output: adapted target domain detector $\mathbf{w}^T$
0: Apply $\mathbf{w}^S$ to $I_t, t \in [1, N]$ and obtain detections $D_s = \{d_s(t)|t \in [1, N]\}$.
1: $\mathbf{w}_c^0 = \mathbf{w}^S$
2: Apply MOT (Alg. 1) on D_s to obtain trajectories $T^*(t), t \in [1, N]$.
3: **for** t=1,2, ..., N, **do**
4: Build the hierarchical model according to $T^*(t)$.
5: Get the positive examples (*i.e.* pedestrians) $d_s^*(t)$ from $T^*(t)$.
6: Extract background examples outside the non-negative region.
7: Optimize the objective function 2 and obtain $\mathbf{w}_c^t$.
8: **end for**
9: $\mathbf{w}^T = \mathbf{w}_c^N$

IV. Experiments

A. Datasets

As source-domain, we use the same virtual-world dataset than [8]. For the target domain, we use three sequences from ETH dataset [26], namely 'BAHNHOF', 'JELMOLI', 'SUNNY DAY', and denoted by ETH0, ETH1, ETH2 respectively. The ETH data was acquired from a "robot perspective" moving in side walks. These sequences are acquired with the same camera but from three different scenarios, that here we consider domains. Since we focus on unsupervised domain adaptation, the fact of using as source domain a virtual-world dataset with automatically provided ground truth, implies that our final adapted pedestrian detector (HOLDA-MOT) has been trained without human annotations.

The experiments are carried out according to Caltech pedestrian detection benchmark [27]. Therefore, we use per-image evaluation, *i.e.*, false positives per image (FPPI) *vs.* miss rate. Following the Caltech benchmark protocol, the detected pedestrians are of height ≥ 50 pixels.

B. HOLDA in unlabeled target domains

In this section, we evaluate the accuracy of the proposed HOLDA-MOT under the fully unsupervised setting, *i.e.*, no target annotations are provided. We present two types of experiments.

In the first type of experiment, we assume that an unlabelled video (target domain) is given to us and we must use it to adapt our current DPM model (learnt with virtual data in our case) without performing human annotation of the objects of interest (pedestrians in this case). Since, a priori, the more training data the better, with this type of experiment we assess if our proposal is able to use as many pedestrians as possible from the target video in order to perform the

TABLE I
EVALUATED DA METHODS

Method	Description	Require labelled data?
SRC	Source domain classifier (no adaptation).	NO
A-SSVM[8]	A batch learning baseline, adaptive SSVM.	YES
INC-MIL[16]	The incremental adaptive DPM based on multiple instance learning.	YES
INT-MIL[16]	The incremental adaptive DPM based on multiple instance learning, with human in the loop.	Limited (Human in the loop)
HOLDA-MOT	The proposed hierarchical online domain adaptation, incorporating multiple object tracking.	NO

domain adaptation. Accordingly, in this case we evaluate the accuracy of our method by comparing the pedestrians detected using our final adapted pedestrian model, with respect to the ones annotated by a human oracle. Of course, considering also false positives, which in the case of the human oracle are zero. We do so for each ETH domain separately.

In the second type of experiment, we evaluate the accuracy of the adapted detector in totally unseen data of the same target domain. For that, given one of the ETH domains (videos) we split it in two parts, *i.e.*, adaptation and evaluation. In this case, we try different splits.

For the first type of experiments, we compare our results to several baselines [16] described in Table I. Note that, INC-MIL and A-SSVM require labeled training data, and around 100 annotated training images are used in these experiments (*i.e.*, reproducing [16]). INT-MIL does not use ground truth but requires a human oracle to click out false positives during the training process. Our algorithm does not require any ground truth information, neither a human oracle in the adaptation loop. The results are shown in Fig. 3. The proposed method HOLDA-MOT improves the source detector more than 10 percentage points on each sequence and approaches the batch learning method A-SSVM, even outperforms A-SSVM on ETH1. Note that A-SSVM version uses all the ground truth provided by the human oracle.

For the second type of experiments, we split the sequence into training and testing sets. For ETH0, because it has more images, we train with 200, 400 and 600 consecutive images and test on the rest. The main goal of this experiment is to investigate the generalization of the adapted detector and to evaluate its accuracy on unseen images. Fig. 3 shows the accuracy for each sequence. As we can see from the results, HOLDA-MOT does the adaptation to unseen images. The portion of unlabeled images does have an impact to the final adapted detector. Around one third of the unlabeled sequence is usually adequate to train the adapted detector.

V. CONCLUSION

In this work, we present an online domain adaptation framework based on the hierarchical adaptation model. The hierarchical model is built on each frame, where leaf nodes are corresponding to pedestrian instance detectors and the root node is corresponding to the pedestrian category detector. The optimization of the hierarchical model is done by a online version of HA-SSVM [9]. The online domain adaptation achieves comparable accuracy to the batch learned models while does not require re-visiting source domain data neither labeled target domain training data. It improves considerably the source classifier too. As the proposed algorithm is general, it could be applied to other SVM-based classifiers as well.

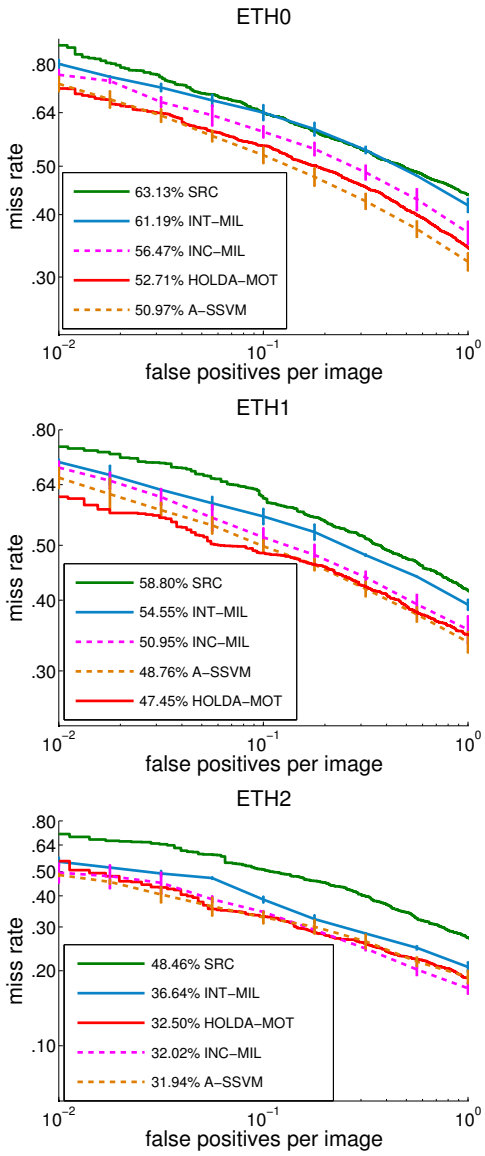

Fig. 2. Comparative results.

ACKNOWLEDGMENT

This work is supported by the Spanish MEC project TRA2014-57088-C2-1-R and DGT project SPIP2014-01352, by the Secretaria d'Universitats i Recerca del Departament d'Economia i Coneixement de la Generalitat de Catalunya (2014-SGR-1506). Our research is also kindly supported by NVIDIA Corporation in the form of different GPU hardware.

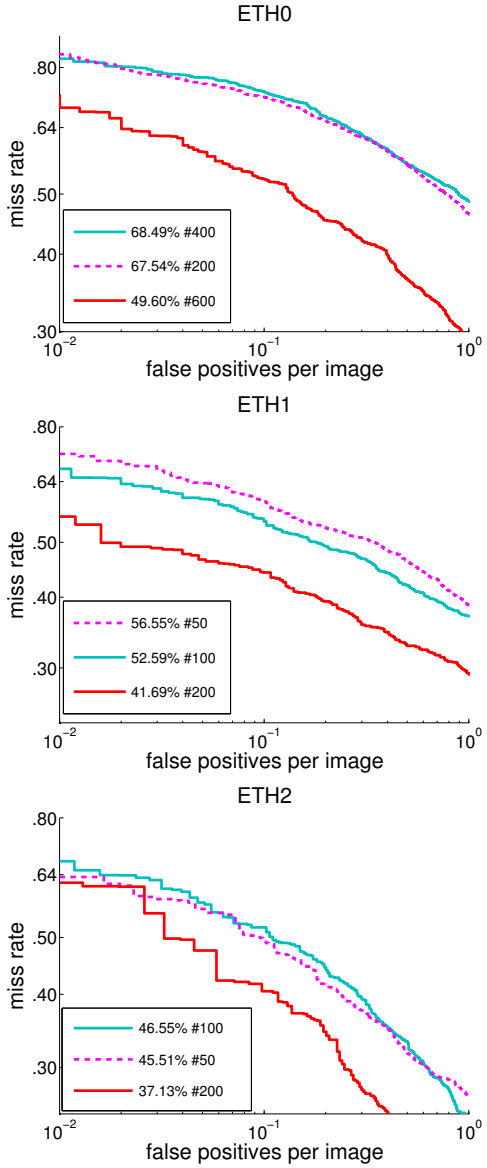

Fig. 3. Testing error of HOLDA-MOT with various train/test splits. ETH0 has 999 images and 1998 pedestrians. ETH1 has 451 images and 902 pedestrians. ETH2 has 354 images and 708 pedestrians.

REFERENCES

[1] A. Khosla, T. Zhou, T. Malisiewicz, A. Efros, and A. Torralba, "Undoing the damage of dataset bias," in *European Conf. on Computer Vision*, Florence, Italy, 2012.

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

[3] 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.

[4] 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.

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

[6] 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.

[7] D. Vázquez, A. López, J. Marín, D. Ponsa, and D. Gerónimo, "Virtual and real world adaptation for pedestrian detection," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 36, no. 4, pp. 797–809, 2014.

[8] J. Xu, S. Ramos, D. Vázquez, and A. López, "Domain adaptation of deformable part-based models," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 36, no. 12, pp. 2367–2380, 2014.

[9] J. Xu, S. Ramos, D. Vázquez, and A. López, "Hierarchical adaptive structural svm for domain adaptation," *Int. Journal on Computer Vision*, 2016.

[10] L. Duan, D. Xu, and I. Tsang, "Learning with augmented features for heterogeneous domain adaptation," in *Int. Conf. on Machine Learning*, 2012.

[11] J. Hoffman, E. Rodner, J. Donahue, K. Saenko, and T. Darrell, "Efficient learning of domain invariant image representations," in *Int. Conf. on Learning Representations*, Arizona, USA, 2013.

[12] J. Jiang, "A literature survey on domain adaptation of statistical classifiers," School of Information Systems, Singapore Management University, Tech. Rep., 2008.

[13] D. Goehring, J. Hoffman, E. Rodner, K. Saenko, and T. Darrell, "Interactive adaptation of real-time object detectors," in *IEEE Int. Conf. on Robotics and Automation*, 2014.

[14] J. Hoffman, S. Guadarrama, E. Tzeng, J. Donahue, R. Girshick, T. Darrell, and K. Saenko, "LSDA: Large scale detection through adaptation," in *Advances in Neural Information Processing Systems*, Quebec, Canada, 2014.

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

[16] J. Xu, S. Ramos, D. Vázquez, A. López, and D. Ponsa, "Incremental domain adaptation of deformable part-based models," in *British Machine Vision Conference*, Nottingham, UK, 2014.

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

[18] B. Babenko, M. Yang, and S. Belongie, "Robust object tracking with online multiple instance learning," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1619–1632, 2011.

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

[20] A. Royer and C. H. Lampert, "Classifier adaptation at prediction time," in *IEEE Conf. on Computer Vision and Pattern Recognition*, Boston, USA, 2015.

[21] D. Hall and P. Perona, "Online, real-time tracking using a category-to-individual detector," in *European Conf. on Computer Vision*, Zurich, Switzerland, 2014.

[22] A. Milan, S. Roth, and K. Schindler, "Continuous energy minimization for multitarget tracking," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 36, no. 1, pp. 58–72, 2014.

[23] J. Munkres, "Algorithms for assignment and transportation problems," *Journal of the Society for Industrial and Applied Mathematics*, vol. 5, no. 1, 1957.

[24] R. Girshick, "From rigid templates to grammars: Object detection with structured models," Ph.D. dissertation, The University of Chicago, Chicago, IL, USA, 2012.

[25] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part based models," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[26] C. Wojek, S. Walk, and B. Schiele, "Multi-cue onboard pedestrian detection," in *IEEE Conf. on Computer Vision and Pattern Recognition*, Miami Beach, FL, USA, 2009.

[27] 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*, vol. 34, no. 4, pp. 743–761, 2012.