

## Research Article

# Domain Adaptation for Pedestrian Detection Based on Prediction Consistency

Yu Li-ping,<sup>1,2</sup> Tang Huan-ling,<sup>2</sup> and An Zhi-yong<sup>1</sup>

<sup>1</sup> Key Laboratory of Intelligent Information Processing, Universities of Shandong (Shandong Institute of Business and Technology), Yantai 264005, China

<sup>2</sup> School of Computer Science and Technology, Shandong Institute of Business and Technology, Yantai 264005, China

Correspondence should be addressed to An Zhi-yong; [azytyut@163.com](mailto:azytyut@163.com)

Received 25 March 2014; Revised 20 May 2014; Accepted 21 May 2014; Published 10 June 2014

Academic Editor: Liang Lin

Copyright © 2014 Yu Li-ping et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Pedestrian detection is an active area of research in computer vision. It remains a quite challenging problem in many applications where many factors cause a mismatch between source dataset used to train the pedestrian detector and samples in the target scene. In this paper, we propose a novel domain adaptation model for merging plentiful source domain samples with scarce target domain samples to create a scene-specific pedestrian detector that performs as well as rich target domain samples are present. Our approach combines the boosting-based learning algorithm with an entropy-based transferability, which is derived from the prediction consistency with the source classifications, to selectively choose the samples showing positive transferability in source domains to the target domain. Experimental results show that our approach can improve the detection rate, especially with the insufficient labeled data in target scene.

## 1. Introduction

Recently, pedestrian detection has received much attention in the computer vision community. An accurate and fast pedestrian detector is especially useful for various vision applications such as surveillance [1, 2], intelligent vehicles [3, 4], and victim detections [5]. The general method for detecting pedestrian is a window-scanning approach, where rectangular windows of variable scales scanning densely on the image are classified as pedestrian or nonpedestrian. The great progress has been made in this field. Dalal and Triggs [6] proposed the grids of histograms of oriented gradient (HOG) descriptors with a linear SVM classifier and showed experimentally that their feature set significantly outperformed existing features sets for human detection. The powerful deformable part models (DPMs) [7] have shown promising results on challenging datasets. In this model, the appearance of each part and the deformation among parts were considered, respectively, and the star model was used to model the deformation. However, it is still a challenge to train a pedestrian detector working reliably because of wide variability in shape and appearance of the human figure.

The problem is further compounded as illumination conditions, resolutions, and viewpoints change. One way to solve this problem is to require an enormous training set to handle the large uncertainty of the human's visual appearance. However, it is often impractical to collect such a large training set. Moreover, a large number of experiments show that a pedestrian detector trained using a general dataset may suffer a large performance drop in a specific target scene. For example, a general pedestrian detector trained from INRIA dataset is evaluated on the Caltech pedestrian dataset, and the absolute performance is quite unsatisfied with a log-average miss rate of over 80% [4]. This is due to the mismatch between the training INRIA set and the testing Caltech pedestrian dataset on illumination, resolutions, and viewpoint. In fact, the variation is relatively smaller in a specific scene especially in surveillance scene, since videos captured from fixed cameras only have limited variations on viewpoints, resolutions, and backgrounds. Therefore, it is easier to train a scene-specific pedestrian detector with high quality using samples captured from the target scene. The obvious solution would be to retain data from the relevant target scene; however, this often leads to overfitting because

the target data are expensive to label and thus their quantities are limited. How to train an accurate detector using a handful of data from generic scenes and scarce target data is a critical problem.

Techniques for addressing learning problems with mismatched distributions are often referred to as domain adaptation or transfer learning. In this area, the source domains refer to the labeled samples from generic scenes, while the target domain refers to the data from target scene. In addition, it is always assumed that the source and target tasks are the same, while the source and target domains are different. In general, data from source and target domains are semantically related to each other in the sense. That is, the certain parts of data in the source domains can still be reused as additional training data for learning the target model. This learning technology is especially referred to as instance transfer [8].

Approaches to domain adaptation can be divided into two cases: one case is from only one source domain transferring to the target domain; another case is from multiple source domains to the target domain. In this paper, we try to investigate the latter case. For the former, the intuitive idea is to weight a source sample based on the similarity between it and the  $K$  nearest target samples directly. However, a source domain is usually a mixture of several components. In this case, the distribution difference among multiple components is not exploited, but this information is very important to derive the common nature of the multiple source components.

In this paper, we propose a novel boosting-based domain adaptation method to select data from multiple source domains and learn the target model for pedestrian detection. TrAdaBoost [9] is the first algorithm of boosting to domain adaptation, and then TransferBoost [10] is proposed as an improvement over it. TransferBoost algorithm introduces a task-based notion, namely, transferability, into standard Adaboost to weight the source samples, and the transferability of a source domain is defined as the difference in weighted errors on the target domain between learning with and without this source domain transfer. However, in this paper the distribution difference among multiple source domains is stressed, and then an instance-based transferability is given. Distributions of source domains are different but related to each other; we may discover the common nature from multiple source domains. This common nature belongs to all source domains and is helpful for learning the target model. In our model, given  $m$  source domains, we train  $m$  source classifiers, respectively, on each of these source domains. Then each source instance is reweighted and transferred into the target domain according to the degree of consistency in these predictions by multiple source classifiers. That is, the source instance will be granted the maximum weight if all the source classifiers reach a perfect agreement on these prediction results. Similarly, the source instance will be granted the minimum weight if all the source classifiers get opposite opinions.

Finally, following a careful theoretical analysis, we validate the proposed method in the context of scene-specific pedestrian detection. The experimental results show that our

approach can improve the detection rate, especially with the insufficient labeled data in target scene.

*Organization of the Paper.* The following sections are organized as follows. In Section 2, we review the previous work on domain adaptation and scene-specific pedestrian detection. In Section 3, we present our method and parameters definition in detail. We report experimental results in Section 4. And, finally, some brief conclusions are given in Section 5.

## 2. Related Work

Domain adaptation has received much attention in recent years. There is an extensive literature on this subject. Except for the above TrAdaBoost and TransferBoost algorithm, Cao et al. [11] studied the domain adaptation problem for face verification, combining a KL-divergence based prior knowledge with a likelihood function. Shi and Sha [12] defined an information-theoretic metric which balanced two forces: maximizing domain similarity and minimizing the expected classification error on the target domain. Chen et al. [13] combined cotraining learning with domain adaptation and formulated a single optimization problem which simultaneously learned a target predictor and a subset of source and target features.

Another closely related task is detecting scene-specific pedestrians, and existing research on this area is limited compared with extensive literature on generic pedestrian detections. Wang et al. [2] proposed a transfer learning framework to adapt a generic pedestrian detector to a specific scene. They weighted the source samples by exploring the indegrees from target samples to source samples on graph. Moreover, they took the context cues from motions, scene structures, and scene geometry to select samples from the target scene. However, motion based detection is not robust due to many factors, such as shadows and illumination change. Wu and Nevatia [14] presented an unsupervised, online retaining approach based on online boosting, and the oracle (labeler) for unsupervised learning was based on combination of a set of part detectors learned by offline boosting. The main issue of unsupervised learning is a labeler's errors and the selected target samples are likely to have wrong label or inaccurate position or size which may make the detector drift.

## 3. Domain Adaptation Algorithm

In this section, we first define the problem of domain adaptation from multiple source domains to the target domain, and then the framework of domain adaptation algorithm is proposed. Finally, how to choose parameters is introduced.

*3.1. Problem Definition.* Let  $S_1, \dots, S_m$  be  $m$  ( $m > 1$ ) source domains with labeled instances, and the  $i$ th source dataset is represented as  $S_i = \{(x_j, y_j)\}_{j=1}^{|S_i|}$ , where  $y_j = \pm 1$  is the label of instance  $x_j$ . These data from source domains are assumed to be abundant. The target domain is denoted by  $T = \{(x_j, y_j)\}_{j=1}^n$ , where  $n$  is the number of instances in the target domain. The quantity of the target data is often

**Input:**  
 Source domains  $S_1, \dots, S_m$ , where  $S_i = \{(x_j, y_j)\}_{j=1}^{|S_i|}$ , target domain  $T = \{(x_j, y_j)\}_{j=1}^n$ ,  
 The maximum number of iterations  $K$  and a base learning algorithm  $\phi$ .

- (1) Total training data:  $D = S_1 \cup S_2 \cup \dots \cup S_m \cup T$ .
- (2) Initialize the weight  $w_1(x_j) = 1/|D|$ , for  $(x_j, y_j) \in D$ .
- (3) for  $t = 1$  to  $K$  do
  - (3.1) Train a weak hypothesis  $h_t : X \rightarrow Y$  on  $D$  with weights  $w_t(D)$ .
  - (3.2) Compute the weighted error  $\epsilon_t$  of  $h_t$  on  $D$ .  

$$\epsilon_t = \sum_{(x_j, y_j) \in D} w_t(x_j) \mathbb{I}[h_t(x_j) \neq y_j]$$
, where  $\mathbb{I}[\cdot]$  is an indicator function.
  - (3.3) Choose  $\alpha_t \in \mathbb{R}$ .
  - (3.4) for  $i = 1$  to  $m$  do
    - (3.4.1) Train a strong classifier  $H_t^i$  on  $S_i$  with weights  $w_t(S_i)$  based on standard Adaboost algorithm.
  - (3.5) Choose  $\beta_t(x_j) \in \mathbb{R}$ ,  $(x_j, y_j) \in S_1 \cup S_2 \cup \dots \cup S_m$  for based on decision consistency.
  - (3.6) Update the weights for all  $(x_j, y_j) \in D$ :  

$$w_{t+1}(x_j) = \begin{cases} \frac{w_t(x_j) \exp(-\alpha_t y_j h_t(x_j) + \lambda \beta_t(x_j))}{Z_t} & (x_j, y_j) \in S_1 \cup S_2 \cup \dots \cup S_m \\ \frac{w_t(x_j) \exp(-\alpha_t y_j h_t(x_j))}{Z_t} & (x_j, y_j) \in T \end{cases} \quad (*)$$
 where  $Z_t$  normalizes  $w_{t+1}(D)$  to be a distribution.
  - (3.7) end for

**Output:** the hypothesis  $H(x) = \text{sign}(\sum_{t=1}^K \alpha_t h_t(x))$ .

ALGORITHM 1: The proposed domain adaptation algorithm.

insufficient to learn a good model for the target domain. Under the assumption that the distributions of  $S_1, \dots, S_m$  and  $T$  are different but related to each other, the goal of domain adaptation is to train a classifier  $f$  for the target domain, selecting instances from  $S_1, \dots, S_m$  into the target dataset  $T$  in learning the model.

**3.2. Domain Adaptation Algorithm.** Our proposed domain adaptation model is summarized in Algorithm 1, which is based on TransferBoost [11] algorithm. First, data from multiple source domains and the target domain are merged into the whole training dataset  $D$ . Then, instances from both multiple source domains and target domain are reweighted iteratively in different methods. On each iteration  $t$ , given instances from training dataset  $D$  with a distribution  $w_t(D)$ , the base learning algorithm computes a weak hypothesis  $h_t$ ; then, for the purpose of minimizing the weighted error  $\epsilon_t$ ,  $\alpha_t$  is computed and the details are provided in Section 3.4. Moreover, source classifier  $H_t^i$   $i = 1, \dots, m$  is trained individually on source domain  $S_i$  with distribution  $w_t(S_i)$  according to the standard Adaboost paradigm. For each source instance,  $\beta_t(x_j)$  is computed according to the transferability to the target domain and detailed in Section 3.3. Furthermore, the instance is reweighted in this model as follows. The weight of instance in the target domain  $T$  is increased due to whether this instance is mispredicted by weak hypothesis  $h_t$  following the standard Adaboost algorithm; however, each instance in the source domains increases or decreases its weight by two terms in (\*) of Algorithm 1. The first term ensures the

mispredicted instances with higher weight in a similar way, and then the second term  $\beta_t(x_j)$  denotes the transferability from the source instance  $x_j$  to the target domain  $T$ . Our intuition in this model is that the source instances with a high agreement predicted by all the source classifiers could yield better transferability across domains. The coefficient  $\lambda$  controls the balance between these two terms. After several iterations, the final classifier is constructed by a weighted sum of the weak classifiers.

**3.3. Choosing the  $\beta_t(x_j)$  Parameter.** In instance transfer, the instance which shows positive transferability should be transferred into the target domain with higher weight, whereas the instance with negative transferability should avoid to be transferred. Thus, our challenge is how to define the transferability from a source instance to the target domain.

To address this challenge, we first describe how to measure the consistent degree of prediction by multiple source classifiers. From an information-theoretic perspective, the Shannon entropy, which quantifies the uncertainty in a random variable, may be used as a useful candidate metric for consistency. Given a set of classifiers  $\{H^i\}_{i=1}^m$ , each of which may assign confidence score to an instance  $x_j$ , we get a probability distribution vector  $P^i(x_j)$ . The average probability distribution vector may be calculated as

$$\bar{P}(x_j) = \frac{\sum_{i=1}^m P^i(x_j)}{m} \quad (1)$$



FIGURE 1: Some samples and example images (cropped) from four datasets.

Moreover, based on the Shannon entropy, a definition to measure the consistent degree of prediction by multiple classifiers is given by

$$C(P^1(x_j), \dots, P^m(x_j)) = -E(\bar{P}(x_j)), \quad (2)$$

where  $E(\cdot)$  denotes the Shannon entropy and  $\bar{P}(x_j)$  is the average probability distribution vector defined in (1).

Intuitively, the source instant  $x_j$  will show positive transferability if all the source classifiers reach a high agreement on these prediction results. Similarly, the source instant  $x_j$  will show negative transferability if all the source classifiers get opposite opinions. Consequently, transferability from a source instance  $x_j$  to the target domain in iteration  $t$  is described using parameter  $\beta^t(x_j)$  as

$$\beta^t(x_j) = C(P^1(x_j) + \dots + P^m(x_j)) = -E(\bar{P}(x_j)). \quad (3)$$

**3.4. Choosing the  $\alpha_t$  Parameter.** Following TransferBoost algorithm, the following bound holds on the training error  $\varepsilon_T = (1/|T|) \sum_{x_j \in T} \varepsilon_j$ :

$$\varepsilon_T \leq \frac{|D|}{|T|} \prod_{t=1}^K Z_t, \quad (4)$$

where  $\varepsilon_j = \mathbb{I}[H(x_j) \neq y_j]$  ( $(x_j, y_j) \in T$ ) is the error of the instance  $(x_j, y_j)$  suffered by the classifier  $H$ ,  $K$  is the

maximum number of iterations, and  $Z_t$  is a normalization factor. To minimize the bound, we choose the following parameter:

$$\alpha_t = \frac{1}{2} \ln \left( \frac{1 + \sum_{x_j \in D} w_t(x_j) y_j h_t(x_j)}{1 - \sum_{x_j \in D} w_t(x_j) y_j h_t(x_j)} \right). \quad (5)$$

## 4. Experiments

For evaluating our domain adaptation model, we employed two generic-scene datasets (MIT CBCL [15] and INRIA [16]) and two specific-scene datasets (MIT\_Traffic [17] and CAVIAR [18]), respectively. Some pedestrian samples and example images are shown in Figure 1. Decision stump is used as base learning algorithm and HOG descriptor is adopted as the feature representation for the whole test. The false positive per image (FPPI) versus recall rate is used as the evaluation metric.

**4.1. Datasets.** The one generic-scene dataset is obtained from the MIT CBCL dataset, where the pedestrian instances are less varied in pose and illumination and in front and rear viewpoint. 924 pedestrian samples (1848 with their left-right reflections) are exacted for training. Another more comprehensive dataset is INRIA dataset. The INRIA pedestrian dataset consists of the training set and the testing set. In

TABLE 1: The descriptions of the datasets for training.

Dataset	Training set	
	Positives	Negatives
MIT_Traffic_DA	Source domains: total of 4833 samples, INRIA, MIT CBCL, and CAVIAR WalkbyShop1cor (118 frames) Target domain: 790 samples, MIT_Traffic (top 210 frames)	
CAVIAR_DA	Source domains: total of 5054 samples, INRIA, MIT CBCL, and MIT_Traffic (top 210 frames) Target domain: 569 samples and CAVIAR WalkbyShop1cor (118 frames)	6000 samples and INRIA
MIT_Only	790 samples and MIT_Traffic (top 210 frames)	
CAVIAR_Only	569 samples and CAVIAR WalkbyShop1cor (118 frames)	
All	Total of 5623 samples and INRIA + MIT CBCL + MIT_Traffic (top 210 frames) + CAVIAR WalkbyShop1cor (118 frames)	

TABLE 2: The descriptions of the datasets for testing.

Target scene	Test set
MIT_Traffic	MIT_Traffic (bottom 100 frames)
CAVIAR	CAVIAR ShopAssistant2cor (165 frames)

our implements, only 1208 pedestrian samples (2416 with reflections) in training set are used. The pedestrians in the INRIA dataset are in various poses, illumination, and viewpoints, and some of them are partially occluded.

The two specific-scenes surveillance datasets, MIT\_Traffic and CAVIAR, are also involved in our tests. MIT\_Traffic dataset is captured at 30 fps from the real street intersection. These video sequences contain more challenges due to perspective deformation. For the training set, 210 frames with 790 pedestrian instances are uniformly sampled from the first part of the video sequences; meanwhile, 100 frames are uniformly exacted from the rest of video sequences for testing. CAVIAR dataset is captured by a stationary camera with a wide angle lens. 118 frames with 569 pedestrian instances are uniformly sampled from WalkbyShop1cor clip for training, and then 165 frames are uniformly sampled from ShopAssistant2cor clip for testing.

Table 1 shows the descriptions for training sets in our experiments. The first two datasets are used to train our domain adaptation model, respectively, in different target domain. The name of the dataset MIT\_Traffic\_DA indicates that MIT\_Traffic dataset is the target domain in domain adaptation, and the other three datasets are used as source domains. The dataset, CAVIAR\_DA, is named in the same way. Furthermore, we also constructed the next two datasets with samples only from the target domain. Finally, the last dataset consists of all 5623 training samples. In the experiments, we also randomly sampled 6000 negative samples from the INRIA's training negative images (10 for each image) as the training negative set. The descriptions for testing set are shown in Table 2. The dataset for testing is named according to the target scene.

**4.2. Experiments Results.** In the experiments, we first evaluated the proposed domain adaptation model when trained on 30%, 60%, and 100% of the source training data. The algorithm *Adaboost(T)*, trained on only the target domain  $T$ , was also involved in this experiment. Figure 2 shows their learning curves on two target scenes, respectively. In the following discussion, when the recall rate is mentioned, we always assume that FPPI = 2. In Figure 2, the results indicate that the recall rate of our model increases with the increase of the number of source data. Particularly, the proposed model significantly outperforms the Adaboost algorithm trained on only the target domain. On the MIT\_Traffic test dataset, detection rate is improved from 18% to 62%, and, on the CAVIAR test dataset, detection rate is improved from 23% to 73%. This is because the quantity of target data is too small to train a high-efficient pedestrian detector. Experiments also show that the proposed approach performs significantly worse on MIT\_Traffic scene. The reason for that may lie in MIT\_Traffic dataset with more perspective deformation and lower resolution compared with CAVIAR dataset.

Furthermore, Figure 3 compares three algorithms: TransferBoost [10] algorithm, linear SVM [6], and DPM [7] in two target scenes, respectively. Table 3 describes the learning methods in detail. This graph shows that our approach, with the higher recall rate, exhibits competitive results compared with the other three algorithms. On the MIT\_Traffic test dataset, compared with TransferBoost, near 7% gain is achieved by our approach, and, on the CAVIAR test dataset, the gain is about 4%. In addition, it outperforms the generic detectors, SVM detector, and DPM detector, trained on both source and target data. It may be due to the fact that proper using of the source data and avoiding negative transferability can improve the generalization power.

## 5. Conclusions and Future Work

In this paper, we proposed a novel domain adaptation model for pedestrian detection in a specific scene especially in video surveillance. The main idea is to transfer the source instances to a target domain using the common nature among

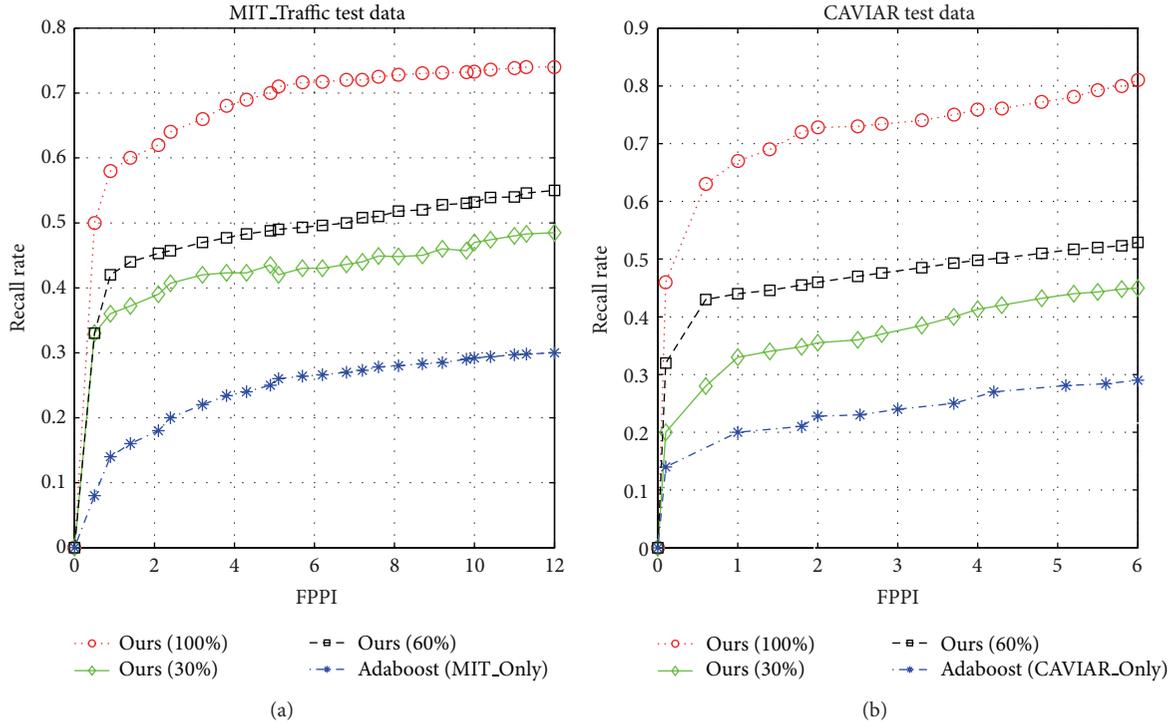


FIGURE 2: Results on MIT\_Traffic scene (a) and CAVIAR scene (b). (a) and (b) compare with Adaboost algorithm trained only on the target domain dataset and our approach trained on different numbers of the source data.

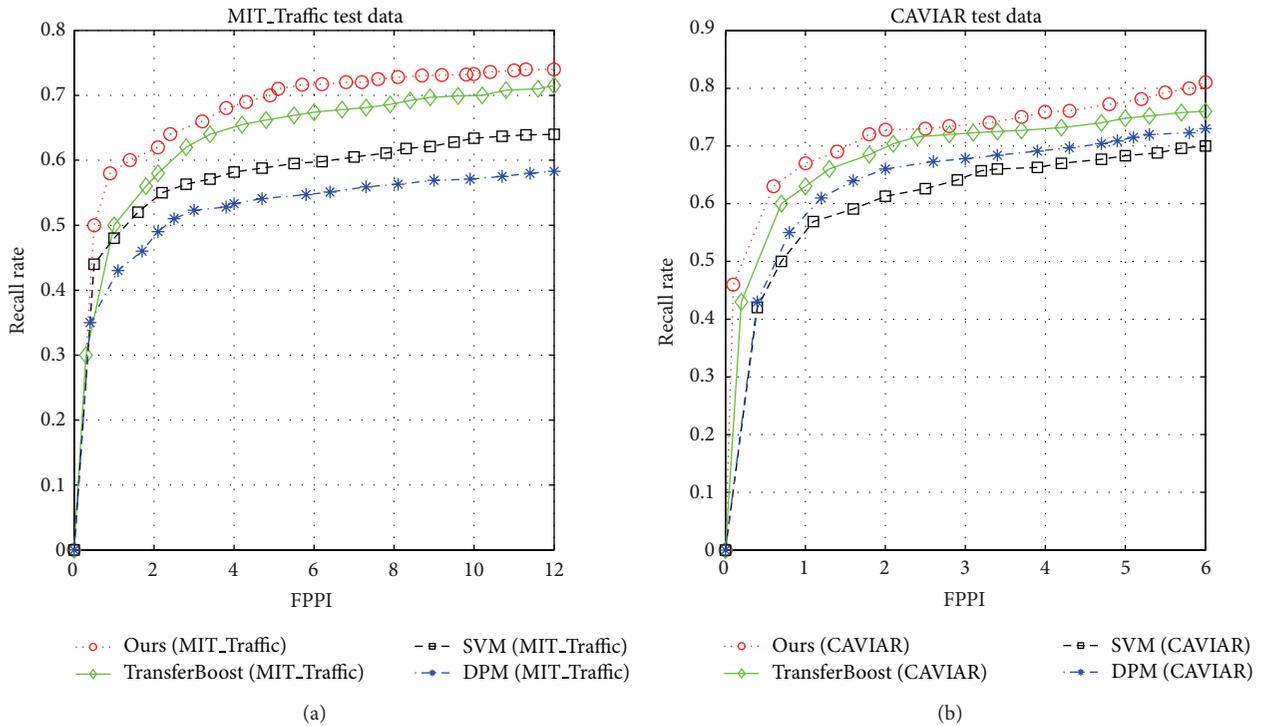


FIGURE 3: Comparison with TransferBoost, linear SVM, and DPM on MIT\_Traffic scene (a) and CAVIAR (b) scene.

TABLE 3: The descriptions of learning methods.

Learning method	Training data	Testing data
Ours (MIT_Traffic)	MIT_Traffic_DA	MIT_Traffic
Ours (CAVIAR)	CAVIAR_DA	CAVIAR
TransferBoost (MIT_Traffic)	MIT_Traffic_DA	MIT_Traffic
TransferBoost (CAVIAR)	CAVIAR_DA	CAVIAR
SVM (MIT_Traffic)	All	MIT_Traffic
SVM (CAVIAR)	All	CAVIAR
DPM (MIT_Traffic)	All	MIT_Traffic
DPM (CAVIAR)	All	CAVIAR

multiple source domains. Accordingly, each source instance is reweighted and transferred into the target domain according to the degree of consistency in the predictions by multiple source classifiers. Moreover, the effectiveness of the proposed algorithm is evaluated through experiments on two video surveillance datasets.

We note that a large number of source instances should be stored when transferring to a specific scene in our approach, and it is inefficient in practical applications. In the future, we will try to extend the model to parameterize the knowledge of multiple source domains.

### Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

### Acknowledgments

This work is supported by National Natural Science Foundation of China (Grant nos. 61175053, 61272244, and 61272430) and Shandong Province Natural Science Foundation (Grant nos. ZR2013FL018 and ZR2011FM035).

### References

- [1] J. K. Aggarwal and M. S. Ryoo, "Human activity analysis: a review," *ACM Computing Surveys*, vol. 43, no. 3, article 16, pp. 1–47, 2011.
- [2] M. Wang, W. Li, and X. Wang, "Transferring a generic pedestrian detector towards specific scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '12)*, pp. 3274–3281, June 2012.
- [3] D. Gerónimo, A. M. López, A. D. Sappa, and T. Graf, "Survey of pedestrian detection for advanced driver assistance systems," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 7, pp. 1239–1258, 2010.
- [4] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: an evaluation of the state of the art," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 4, pp. 743–761, 2012.
- [5] M. Andriluka, P. Schnitzspan, J. Meyer et al., "Vision based victim detection from unmanned aerial vehicles," in *Proceedings of the 23rd IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '10)*, pp. 1740–1747, October 2010.
- [6] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, pp. 886–893, June 2005.
- [7] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.
- [8] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.
- [9] W. Dai, Q. Yang, G.-R. Xue, and Y. Yu, "Boosting for transfer learning," in *Proceedings of the 24th International Conference on Machine Learning (ICML '07)*, pp. 193–200, ACM, June 2007.
- [10] E. Eaton and M. Desjardins, "Selective transfer between learning tasks using task-based boosting," in *Proceedings of the 25th AAAI Conference on Artificial Intelligence (AAAI '11)*, pp. 337–342, August 2011.
- [11] X. Cao, D. Wipf, F. Wen, and G. Duan, "A practical transfer learning algorithm for face verification," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV '13)*, pp. 2887–2894, December 2013.
- [12] Y. Shi and F. Sha, "Information-theoretical learning of discriminative clusters for unsupervised domain adaptation," in *Proceedings of the 29th International Conference on Machine Learning (ICML '12)*, pp. 1079–1086, July 2012.
- [13] M. Chen, K. Q. Weinberger, and J. C. Blitzer, "Co-training for domain adaptation," in *Proceedings of the 25th Annual Conference on Neural Information Processing Systems (NIPS '11)*, pp. 2456–2464, January 2011.
- [14] B. Wu and R. Nevatia, "Improving part based object detection by unsupervised, online boosting," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '07)*, June 2007.
- [15] MIT CBCL, <http://cbcl.mit.edu/software-datasets/PedestrianData.html>.
- [16] INRIA, <http://pascal.inrialpes.fr/data/human/>.
- [17] X. Wang, X. Ma, and W. E. Grimson, "Unsupervised activity perception in crowded and complicated scenes using hierarchical bayesian models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 3, pp. 539–555, 2009.
- [18] Caviar, <http://homepages.inf.ed.ac.uk/rbf/CAVIARDATA1/>.




**Hindawi**

Submit your manuscripts at  
<http://www.hindawi.com>

