Semi-Supervised Domain Adaptation with Instance Constraints

Jeff Donahue[1,2], Judy Hoffman[1,2], Erik Rodner[2,3], Kate Saenko[4], Trevor Darrell[1,2]
[1]UC Berkeley EECS, [2]UC Berkeley ICSI, [3]University of Jena, [4]University of Massachusetts, Lowell
[1]{jdonahue,jhoffman,trevor}@eecs.berkeley.edu, [3]erik.rodner@uni-jena.de, [4]saenko@cs.uml.edu

Abstract

Most successful object classification and detection methods rely on classifiers trained on large labeled datasets. However, for domains where labels are limited, simply borrowing labeled data from existing datasets can hurt performance, a phenomenon known as "dataset bias." We propose a general framework for adapting classifiers from "borrowed" data to the target domain using a combination of available labeled and unlabeled examples. Specifically, we show that imposing smoothness constraints on the classifier scores over the unlabeled data can lead to improved adaptation results. Such constraints are often available in the form of instance correspondences, e.g. when the same object or individual is observed simultaneously from multiple views, or tracked between video frames. In these cases, the object labels are unknown but can be constrained to be the same or similar. We propose techniques that build on existing domain adaptation methods by explicitly modeling these relationships, and demonstrate empirically that they improve recognition accuracy in two scenarios, multi-category image classification and object detection in video.

1. Introduction

Domain adaptation methods are necessary for many real-world applications where test examples differ significantly from the examples used for learning. Prior methods have shown that explicitly modeling and compensating for the domain shift from the source domain to the target (test) domain can significantly boost performance on the target domain. Supervised approaches do this by utilizing a few labeled examples in the target domain [16, 1, 7, 9, 18, 24, 28], while semi-supervised methods also take into account the (typically much more abundant) unlabeled target samples [13, 14].

In many problems, additional *instance* constraints are available over the unlabeled target data, encoding the knowledge that certain samples belong to the same object instance, and thus should be classified in a similar way. To illustrate, consider two practical scenarios. The first (Figure 1) is adapting object detectors trained on static images to a video domain, using a few available labeled videos, and a large number of automatically extracted moving object constraints, e.g. using spatio-temporal segmentation [19]. The second scenario (Figure 2) is adapting multi-category classifiers to a domain where only a subset of categories have (limited) labels, but the same object instances are observed from multiple views/cameras. In both of the above scenarios, unlabeled instance constraints can provide additional information to the classifier about the structure of the target domain, yet such information has not to our knowledge been used for domain adaptation.

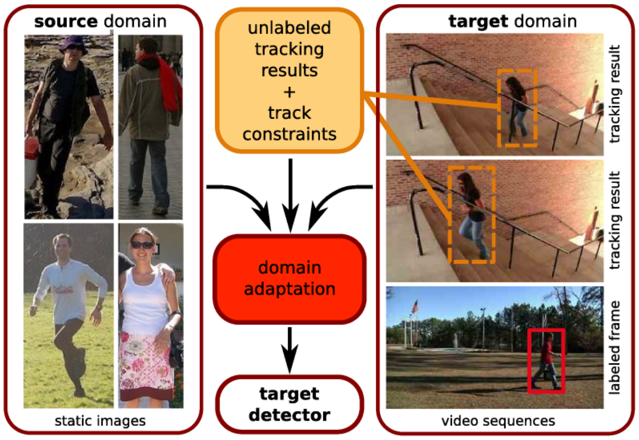


Figure 1. To adapt source classifiers to the target domain, we exploit unlabeled instance constraints in addition to labeled examples. In this case, the source domain is static images, the target is surveillance video, and the instance constraints come from tracking an object between video frames.

In this paper, we present a unified domain adaptation framework that incorporates both traditional labels and unlabeled instance constraints. Our approach is broadly applicable in a range of adaptation settings, including heterogeneous features, detection, classification, and transfer of learned domain shift to unlabeled categories. In particular, we build on a class of adaptive SVM methods that includes the projective model transfer approach of [1] as well as the transform-based approach of [16]. Our main contribution

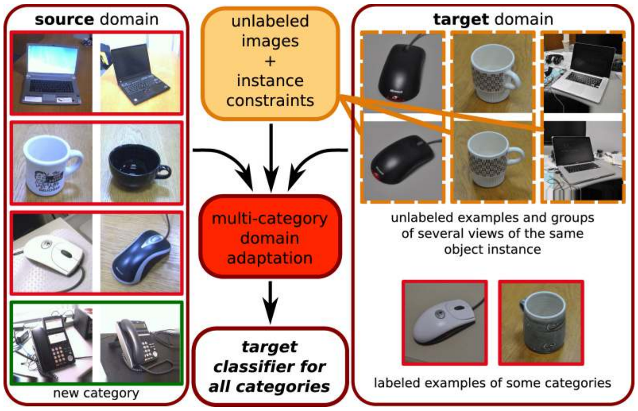

source domain
new category
unlabeled images + instance constraints
multi-category domain adaptation
target classifier for all categories
target domain
unlabeled examples and groups of several views of the same object instance
labeled examples of some categories

Figure 2. Illustration of our approach for multi-category adaptation: a target classifier is learned using labeled images from a subset of categories, plus unlabeled images with instance constraints, which in this case come from images of the same object taken from different views.

is extending these methods to include Laplacian regularization using instance constraints that are encoded by an arbitrary graph. We demonstrate the effectiveness of the algorithm on both detection and classification tasks using publicly available vision datasets. In both cases, our algorithm provides a significant improvement over algorithms with no adaptation and those using adaptation without instance constraints.

Outline The paper is structured as follows: We first review related work in the area of domain adaptation for object recognition and detection in videos (Sect. 2). Our domain adaptation method and the integration of instance constraints is presented in Sect. 3. Furthermore, Sect. 4 and 5 show how to apply the proposed techniques for multi-category and video domain adaption. Experiments in Sect. 6 for object categorization as well as object detection in videos show the benefits of our methods. A summary of our findings and a discussion of future research directions conclude the paper.

2. Related Work

Domain adaptation, or covariate shift, is a fundamental problem in machine learning (see [17] for a comprehensive overview.)

In computer vision, supervised methods based on support vector machines have been popular. These include simple methods such as a weighted combination of source and target SVMs; transductive SVMs applied to adaptation [2]; the feature replication method of [6]; Adaptive SVM [20, 27], where the source model parameters are adapted by adding a perturbation function, and its successor PMT-SVM [1]; The general idea behind Adaptive SVMs is to learn the target classifier $f(\boldsymbol{x})$ as a perturbed version of the existing, source classifier $f^A(\boldsymbol{x})$ via the equation $f(\boldsymbol{x}) = f^A(\boldsymbol{x}) + (\delta f)(\boldsymbol{x})$, where $(\delta f)(\boldsymbol{x})$ is the perturbation function. The drawback is that the learned perturbation cannot be transferred to novel categories, and cannot handle heterogeneous features. In contrast, transform-based supervised methods attempt to learn a perturbation over the feature space rather than a class-specific perturbation over the model parameters, typically in the form of a transformation matrix [16, 9, 18, 24]. Other feature adaptation methods include [5, 11].

Semi-supervised visual adaptation methods have also been proposed, including transform-based methods that use unlabeled data to construct a manifold [13, 14], and SVM-based method that include an unlabeled data term that minimizes the mismatch in the domain distributions, measured by the maximum mean discrepancies [8]. The approaches of [13, 14, 26] can accommodate fully unsupervised domain adaptation.

Instance similarity constraints have been used in multi-view learning [4, 11, 21], canonical-correlation analysis [15], and as constraints *between* domains when available [24]. As far as we know, our approach is the first to utilize such constraints in the target domain. We build on the ideas of Laplacian SVM [22], which requires that the target function vary smoothly on the unlabeled examples.

Tracking-by-detection is a related area of research (e.g. [3]), where an existing classifier reports the initial object location and then is continuously adapted to track the object in the video. In contrast to our work, tracking-by-detection trains a detector specific to the test video, and does not attempt to improve the performance of a generic category detector on novel videos. Learning generic object detectors from both static images and weakly labeled videos is done by [23], where the category label is known but the precise object location in the video is unknown. However, the authors focus on the automatic spatio-temporal segmentation of objects in videos to obtain labeled examples and use a very simple adaptation scheme (weighted combination of source and target). Our method can incorporate spatio-temporal constraints directly over unlabeled examples.

3. Domain adaptation with auxiliary similarity constraints

A popular and effective class of domain adaptation algorithms jointly learns a hyperplane classifier of the source and target domains. We build on this general approach by additionally incorporating constraints obtained from a given similarity graph defined on unlabeled target instances.

We assume we are given labeled source data $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n_s}$ and labeled target data $\tilde{\mathcal{D}}^L = \{(\tilde{\mathbf{x}}_j, \tilde{y}_j)\}_{j=1}^{n_t^L}$, where $n_t^L << n_s$. Additionally we are given unlabeled target data $\tilde{\mathcal{D}}^U = \left\{\tilde{\mathbf{x}}_j^U\right\}_{j=1}^{n_t^U}$ and an edge weight matrix

$\mathbf{B} = (\beta_{j,j'})$ which contains weights for each pair of target training examples. With the unlabeled target data, $\tilde{\mathcal{D}}^U$, and the edge weight matrix, $\mathbf{B}$, we then construct a graph $\mathcal{G} = (\tilde{\mathcal{D}}^U, \mathbf{B})$.

An edge weight, $\beta_{j,j'}$, defines the similarity between two unlabeled target examples and is incorporated to integrate domain the unlabeled examples into domain adaptation.

We first describe our approach and then we demonstrate its generality by integrating it with two specific domain adaptation algorithms.

3.1. Integrating the similarity graph

In the following, we focus on linear models due to their efficiency at test time. This is an essential property for many of the most successful and widely-used approaches to object detection today, which commonly score thousands of bounding boxes per image in a "sliding window" approach to detection at test time [12].

Learning framework Our goal is to learn classifier functions, $f(\mathbf{x}) = \boldsymbol{\theta}^T\mathbf{x}$ for the source and $\tilde{f}(\tilde{\mathbf{x}}) = \tilde{\boldsymbol{\theta}}^T\tilde{\mathbf{x}}$, for the target domain. Many max-margin based domain adaptation optimization techniques can be described generally in terms of the hyperplane parameters, $\boldsymbol{\theta}, \tilde{\boldsymbol{\theta}}$, an optional transformation parameter, A, and loss functions of these parameters and the data. Formally, this can be denoted as follows:

$$\min_{\boldsymbol{\theta},\tilde{\boldsymbol{\theta}},A} \quad \mathcal{R}(\boldsymbol{\theta}, \tilde{\boldsymbol{\theta}}, A) + C \cdot \mathcal{L}(\mathcal{D}, \boldsymbol{\theta}) + \tilde{C} \cdot \tilde{\mathcal{L}}(\tilde{\mathcal{D}}^L, \tilde{\boldsymbol{\theta}}) \quad (1)$$

where $\mathcal{R}$ is a regularizer over all parameters and $\mathcal{L}, \tilde{\mathcal{L}}$ represent the loss terms on the source and target data, respectively. C and $\tilde{C}$ are scalar parameters to be set to trade-off the impact of the source and target data.

For our algorithm we will modify this general formulation to include additional constraints available from the similarity graph, $\mathcal{G}$, available on the unlabeled target data.

Manifold regularization To integrate unlabeled data with similarity constraints into a learning objective function, we use manifold regularization in the form of a Laplacian regularizer, which has been shown effective for semi-supervised learning [22]. This regularizer restricts the function $\tilde{f}(\tilde{\mathbf{x}})$ to have similar values for similar instances.

Given the edge weights $\mathbf{B} = (\beta_{j,j'})$, we can define the Laplacian matrix as $\mathbf{L} = \mathbf{D} - \mathbf{B}$ with $\mathbf{D}$ being the diagonal matrix that contains the row sums of $\mathbf{B}$. Finally, the following function expresses the regularization term that incorporates the similarity constraints over the unlabeled target data.

$$\begin{aligned} r(\tilde{\mathcal{D}}^U, \tilde{f}_{\tilde{\boldsymbol{\theta}}}) &= \frac{1}{2}\tilde{\mathbf{f}}^T\mathbf{L}\tilde{\mathbf{f}} = \frac{1}{2}\left(\tilde{\boldsymbol{\theta}}^T\tilde{\mathbf{X}}^U\right)^T \mathbf{L} \left(\tilde{\boldsymbol{\theta}}^T\tilde{\mathbf{X}}^U\right) \\ &= \sum_{j \neq j'} \beta_{j,j'} \left(\tilde{f}(\tilde{\mathbf{x}}_j^U) - \tilde{f}(\tilde{\mathbf{x}}_{j'}^U)\right)^2 \ , \qquad (2) \end{aligned}$$

where $\tilde{\mathbf{X}}^U$ denotes the matrix containing the unlabeled target training examples as columns and $\tilde{\mathbf{f}} = \tilde{\boldsymbol{\theta}}^T\tilde{\mathbf{X}}^U$. This regularization term can be added to the general formulation from Eq. (1) to produce a unified optimization framework, which can utilize both labeled examples from the target and unlabeled examples that have auxiliary similarity information. This can be seen as a generalization and extension of the semi-supervised approach of [22] to the domain adaptation setting.

3.2. Domain adaptation models

For concreteness we next present our full optimization framework applied to two separate semi-supervised domain adaptation algorithms.

Projective model transfer SVM (PMT-SVM) The PMT-SVM method of [1] assumes that the source hyperplane $\boldsymbol{\theta}$ is given and was learned on the source dataset $\mathcal{D}$ with examples $\mathbf{x}$ in the same feature space as examples $\tilde{\mathbf{x}}$ in the target dataset $\tilde{\mathcal{D}}$. In fact, the key idea of PMT-SVM with respect to domain adaptation is the adaptation regularizer that couples the target and the given fixed source hyperplane using the angle $\alpha(\tilde{\boldsymbol{\theta}}, \boldsymbol{\theta}) = \cos^{-1}\left(\frac{\tilde{\boldsymbol{\theta}}^T\boldsymbol{\theta}}{\|\tilde{\boldsymbol{\theta}}\|\|\boldsymbol{\theta}\|}\right)$ between them. We can express this regularizer in terms of our general framework as:

$$\mathcal{R}^{\text{pmt}}(\boldsymbol{\theta}, \tilde{\boldsymbol{\theta}}) = \frac{1}{2}\|\tilde{\boldsymbol{\theta}}\|_2^2 + \frac{\Gamma}{2}\|\tilde{\boldsymbol{\theta}}\|_2^2 \ \sin^2\alpha(\tilde{\boldsymbol{\theta}}, \boldsymbol{\theta}) \ . \qquad (3)$$

The hyperplane parameters are further restricted to have non-negative correlation $\boldsymbol{\theta}^T\tilde{\boldsymbol{\theta}} \geq 0$ [1]. The second term of the regularizer results in low values when the source and target hyperplane parameters are similar in terms of angular distance, which directly models the main assumption of domain adaptation that both domains share common properties and relevant features. For the loss term $\tilde{\mathcal{L}}$, a standard hinge loss is used, resulting in a modified SVM optimization problem.

Max-margin domain transforms (MMDT) The idea of MMDT, a transform-based domain adaptation approach proposed by [16], is to find a transformation $\mathbf{A}$ between the target and source domains allowing for joint learning of the classifiers in both domains. This allows for complex domain shifts that cannot be modeled by PMT-SVM. MMDT jointly learns a direct transformation together with hyperplane parameters in the source domain. Therefore, we implicitly define $\tilde{\boldsymbol{\theta}} = A^T\boldsymbol{\theta}$ such that the final latent function of the target domain is modeled by $\tilde{f}(\tilde{\mathbf{x}}) = \tilde{\boldsymbol{\theta}}^T\tilde{\mathbf{x}} = \boldsymbol{\theta}^T\mathbf{A}\tilde{\mathbf{x}}$ and the optimization of Eq. (1) is done with respect to $\boldsymbol{\theta}$ and $\mathbf{A}$. Following [16], we use a standard ℓ_2 regularizer for the source hyperplane and a Frobenius norm regularizer for the transformation leading to an over-all regularizer function as follows:

$$\mathcal{R}^{\text{trans}}(\boldsymbol{\theta}, \mathbf{A}) = \frac{1}{2}\|\boldsymbol{\theta}\|_2^2 + \frac{\gamma}{2}\|\mathbf{A} - \mathbf{I}\|_F^2 \ . \tag{4}$$

Note that the transformation is regularized with respect to the identity matrix so that in the case of large values of γ a classifier using source and target data will be learned.

We compute both loss terms with the hinge loss function $H(z) = \max(0, 1 - z)$:

$$\mathcal{L}(\mathcal{D}, \boldsymbol{\theta}) = \sum_{i=1}^{n_s} H(y_i \cdot \boldsymbol{\theta}^T x_i)$$

$$\tilde{\mathcal{L}}(\tilde{\mathcal{D}}^L, \tilde{\boldsymbol{\theta}}) = \sum_{j=1}^{n_t^L} H(\tilde{y}_j \cdot \tilde{\boldsymbol{\theta}}^T\tilde{\mathbf{x}}_j) = \sum_{j=1}^{n_t^L} H(\tilde{y}_j \cdot \boldsymbol{\theta}^T\mathbf{A}\tilde{\mathbf{x}}_j)$$

3.3. Optimization details

We incorporate similarity constraints into the PMT-SVM and transform-based domain adaptation (Sect. 3.1) by adding the additional regularization terms of Eq. (2), incorporating the unlabeled data into the objective functions:

$$\min_{\boldsymbol{\theta}, \tilde{\boldsymbol{\theta}}, A} \quad \mathcal{R}(\boldsymbol{\theta}, \tilde{\boldsymbol{\theta}}, A) + \mathcal{L}(\mathcal{D}, \boldsymbol{\theta}) \\ + \tilde{\mathcal{L}}(\tilde{\mathcal{D}}^L, \tilde{\boldsymbol{\theta}}) + r(\tilde{\mathcal{D}}^U, \tilde{f}_{\tilde{\boldsymbol{\theta}}}). \tag{5}$$

The Laplacian term in (2) is convex in the parameters, which is important for a robust optimization with gradient-based optimization techniques and suitable convergence properties. The extended PMT-SVM approach is optimized using stochastic gradient descent. In contrast to PMT-SVM, the transform-based approach is not convex in all parameters but in both subproblems assuming either a fixed transformation $\mathbf{A}$ or a constant source hyperplane $\boldsymbol{\theta}$. Therefore, learning is done by optimizing with respect to $\mathbf{A}$ and $\boldsymbol{\theta}$ independently of each other until convergence.

4. Multi-category adaptation

Next we consider the setting where we have labeled examples for multiple categories in the source domain, and very few or no examples of some categories in the target domain. Additionally, for the unlabeled data in the target domain, we assume instance constraints are available. Since these constraints are not connected to any labeled examples, we need to use our semi-supervised framework to create a multi-category target classifier. The weights, $\beta_{j,j'}$, for the Laplacian regularizer, can be used to encode confidence about the similarity of two particular images showing the same object instances from different views.

As shown by [16], MMDT can easily be extended to multiple classes, even when not all classes have labeled training data. The key idea is to learn hyperplane parameters $\boldsymbol{\theta}^{(k)}$ for each category k in a one-versus-all manner, which can be formulated in a multi-task manner where the transformation $\mathbf{A}$ is a shared parameter. Specifically, the final optimization objective function will be a sum over classes of class-specific regularizers and loss functions:

$$\min_{\{\boldsymbol{\theta}^{(k)}\}, \mathbf{A}} \quad \sum_k \Big(\mathcal{R}^{\text{trans}}(\boldsymbol{\theta}^{(k)}, \mathbf{A}) + \mathcal{L}(\mathcal{D}, \boldsymbol{\theta}^{(k)}) \\ + \tilde{\mathcal{L}}(\tilde{\mathcal{D}}^L, \mathbf{A}^T\boldsymbol{\theta}^{(k)}) + r(\tilde{\mathcal{D}}^U, \tilde{f}_{\tilde{\boldsymbol{\theta}}_k}) \Big) \ . \tag{6}$$

5. Video domain adaptation

We consider a setting in which we have an object detector (source detector) trained on a source domain, and we would like to perform detection on a target dataset consisting of videos. In particular, we adapt a filter-based object detector such as the deformable parts model (DPM) [12] to a video corpus in which we exploit the signal in the temporal structure of the video data by imposing similarity constraints on the adapted detector. Our method assumes that a small subset of the target video dataset has labeled bounding boxes, and another subset has *unlabeled* bounding boxes with tracks, for example from background subtraction or from another automatic approach, such as [23]. A single track traces the motion path of a single object instance (e.g., a particular person) throughout frames of a video by indicating in each frame which bounding box, if any, corresponds to the object instance.

Formally, assume that we are given an SVM parameter vector $\boldsymbol{\theta}$ trained on labeled source data $\mathcal{D}$. Furthermore, we have a set of video target data with a few videos with some labeled frames. For each unlabeled example $\tilde{\mathbf{x}}_j^U$, we also have a timestamp or frame number $\tilde{r}_j$ as well as a track ID $\tilde{t}_j$ indicating to which track the object belongs. Two examples $\tilde{\mathbf{x}}_j^U$ and $\tilde{\mathbf{x}}_{j'}^U$ belong to the same track if and only if $\tilde{t}_j = \tilde{t}_{j'} \neq 0$, letting $\tilde{t}_j = 0$ denote that example $\tilde{\mathbf{x}}_j^U$ does not belong to any track.

We then train a PMT-SVM with the similarity constraint version of the objective function. In the case of video, we use the Laplacian constraint to reflect the intuition that examples from the same track in nearby frames should have similar classifier outputs. Hence, in our Laplacian term we give positive similarity weightings $\beta_{j,j'}$ to such pairs of examples $\tilde{\mathbf{x}}_j^U, \tilde{\mathbf{x}}_{j'}^U$ and dampen the weighting as the frame distance $|\tilde{r}_j - \tilde{r}_{j'}|$ grows. In particular, we set the following weights for $j \neq j'$:

$$\beta_{j,j'} = \begin{cases} \frac{1}{|\tilde{r}_j - \tilde{r}_{j'}|} & \text{if } \tilde{t}_j = \tilde{t}_{j'} \neq 0, 0 < |\tilde{r}_j - \tilde{r}_{j'}| \leq \tau \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

Domain	# Images available	# Labeled Per Category	# Categories with Labeled Ex.
`webcam`	795	8	31
`dslr`	498	1	16

Table 1. *Office* dataset description [18].

The parameter τ specifies the maximum frame distance between examples for which a similarity constraint is used, and may be set to a finite value to control the number of similarity constraints considered in the optimization problem. Note that the definition of the weights leads to a sparse weight matrix $\mathbf{B}$, which is important for allowing learning to be fast with a large number of training examples.

6. Experiments

In the following, we evaluate our approach in two different scenarios where similarity constraints can easily be exploited: multi-category classification with instance-level constraints and video domain adaptation of a pedestrian detector.

6.1. Multi-category classification

Dataset We evaluate our algorithm in a multi-class classification setting using the *Office* benchmark domain adaptation dataset of [18]. This dataset offers two domains available with instance constraints (`webcam`, `dslr`), see Table 1. The `webcam` domain is a collection of objects in an office environment taken with a webcam. Similarly, the `dslr` domain is a collection of the same objects taken with a DSLR camera. Therefore, there is a resolution and lighting domain shift. We explore the setting where most of the available training data is from a source domain (`webcam`) that is misaligned with the test data that is drawn from the target domain (`dslr`).

The dataset is available with precomputed SURF bag of words feature vectors quantized to 800 dimensions. Following [13] we first apply PCA to the source and target data and then use the lower dimensional data as input to our method and all baselines.

Experimental setup and baselines We assume that there is very little labeled data available from the target domain. For our experiments we only allowed one labeled example per category for each of the first 16 categories. This means that there are a total of only 16 labeled examples available in the target domain. The other 15 categories have labeled data available from only the source domain. Thus, this experiment demonstrates that the algorithm is able to generalize to categories without labels, using only similarity constraints.

To evaluate the performance of our algorithm we compare against a number of baselines that are described below.

svm$_S$	**svm**$_{S\cup T}$	**da only**	**da + lap-sim**
45.80 ± 2.2	50.79 ± 2.4	54.57 ± 2.2	$\mathbf{56.15 \pm 2.7}$

Table 2. Classification results over target test data using the *Office* dataset. Only one labeled example is available from only half of the categories in the target domain. The rest of the target data is assumed to have similarity constraints which can be used by the full similarity constraint algorithm.

svm$_S$ A standard Support Vector Machine classifier trained using only the source data.

svm$_{S\cup T}$ A standard Support Vector Machine classifier trained using both the source and labeled target data.

da only Uses source and labeled target data to train a semi-supervised domain adaptation model using the MMDT optimization from Sect. 3.3. This is equivalent to setting weights $\beta_{j,j'} = 0$ in our model.

Additionally, we show results for our proposed extension of MMDT, denoted as **da + lap-sim**, for domain adaptation with Laplacian regularization. Please note that the PMT extension is not suitable for the multi-class experiment, because it does not allow for generalization to new categories.

Algorithm hyperparameters for all methods were set by cross-validation of the **da only** baseline.

Results and analysis We ran the classification experiment on 10 random train/test splits of the data (as was done in the previous use of this dataset [18, 13]) and we present the average results across the runs in Table 2. We found that in this setting, with only a small amount of target training data, the transform-based domain adaptation method was able to learn to successfully adapt to the target.

Additionally, we found that adding the similarity constraints from the unlabeled target data resulted in a significant performance increase. The Laplacian regularization explicitly optimizes the classifier scores of the same instances to be similar, which added further constraints that aided in adapting the final target classifier. This experiment validates our claim that adding the unlabeled instance constraints from the target can boost performance of a semi-supervised domain adaptation method.

To further evaluate the effect of adding the similarity constraints we also report the multi-class accuracy for only the categories that have no labeled target training data in Table 3. It is interesting to note that domain adaptation alone does not dramatically improve the results on the novel target categories. This is to be expected since there is very little training data from the first 16 categories from which to learn a transformation that generalizes to new categories. However, with the similarity constraints added, our full model dramatically improves on the novel target categories. This again validates our argument that the auxiliary similarity

svm$_S$	**svm**$_{S\cup T}$	**da only**	**da + lap-sim**
35.46 ± 1.0	34.15 ± 0.9	35.79 ± 1.4	$\mathbf{39.89 \pm 1.3}$

Table 3. Classification results for only the target test data from categories with no labeled target training data. Further analysis of experiment from Table 2.

Domain/Dataset	# Images / Frames Labeled
PASCAL VOC 2007 [10]	5011
VisInt [25]	$\{10, 20, 30, 40\}$

Table 4. *PASCAL* to *VisInt* dataset description

constraints can be used in conjunction with a domain adaptation algorithm to learn a more generalizable target model.

6.2. Object detection in video

We now present results showing that similarity constraints can be used to capture useful information about relationships among examples from the same track, which can be leveraged to significantly improve the performance of a domain adapted detector in video. Our experiments focus on person detectors, due to the wide interest in and broad applications of pedestrian detection. The source domain has images from the PASCAL VOC 2007 dataset [10], and the target domain consists of frames of the videos from the VisInt dataset [25].

Experimental setup and baselines In training both the source and target models, we mainly follow the training protocol of the deformable parts model (DPM) [12], which we briefly summarize here.

Due in part to the very large number of negative examples in a detection dataset, the DPM uses bootstrapping methods to optimize training a detector with little compromise on performance. The first training iteration uses warped positives (the original labeled bounding boxes warped to the correct aspect ratio) and random negatives (random subwindows of images with no positive windows). Once an initial model is trained, all subsequent training iterations use *latent positives* and *hard negatives* as the training set. Latent positives are computed by finding the optimal configuration of latent variables (including component selection and placement of root and part filters) with respect to a particular positive bounding box input, given the current detector model. Hard negatives are the highest scoring bounding boxes from images in the training set with no positive bounding boxes.

The source domain detector is trained as in [12], with the exceptions that we use only a single component (with left- and right-facing versions) and do not use any parts. Our target domain training protocol, on the other hand, differs somewhat more significantly from that of the base DPM. Rather than initializing an "empty" model with $\boldsymbol{\theta} = 0$, the target model is instead initialized to the source model, allowing us to skip to the latent positive and hard negative phases of training immediately. Both labeled and unlabeled bounding boxes are used in the latent positive stage of training. Latent positives found from a labeled bounding box are added as regular examples. For unlabeled bounding boxes, the latent positives $\tilde{\mathbf{x}}_j, \tilde{\mathbf{x}}'_j$ found for any bounding box pairs with $\beta_{j,j'} > 0$ are added to the set of constraints (but not as labeled examples). To account for the relatively high proportion of images with a positive bounding box in the VisInt dataset, we also allow hard negatives to be harvested from images with positive bounding boxes, still disallowing any overlap with a positive bounding box. We train using a PMT-SVM with the instance constraints described above (Sect. 3.1), rather than a standard SVM as used by the DPM. In each training iteration, the original source model is used as the source parameter vector $\boldsymbol{\theta}^s$ in the PMT-SVM.

The source detector used in all domain adaptation detection experiments is trained on the *train+val* portion of the PASCAL 2007 dataset. In each experiment we choose a total of N_f labeled frames and $5N_f$ unlabeled frames. Each video contributes exactly 10 frames, which are sampled at an interval of 10 apart. Hyperparameters $(\tilde{C}, \Gamma, \beta)$ are chosen by cross-validation on held-out data.

To evaluate our domain adaptation algorithm with similarity constraints, we compare against the following baselines. All baseline detectors are also trained in the DPM [12] with the protocol described in Sect. 5. Following the experimental setup of [1], we do not use any parts.

dpm$_S$ DPM trained using only the source data.

dpm$_{S\cup T}$ DPM trained using both the source and labeled target data.

da only DPM with domain adaptation trained using source and labeled target data model using the PMT-based optimization from Sect. 3.1. This is equivalent to setting weights $\beta_{j,j'} = 0$ in our model.

Results and analysis Table 5 shows our results for $N_f = 10, 20, 30, 40$, and Figure 3 shows the precision-recall curves. With a single video ($N_f = 10$) of labeled training data, our results show that our method of integrating similarity constraints with domain adaptation (**da + lap-sim**) gives a 3.0% improvement over direct application of the source detector (**dpm**$_S$). With two videos ($N_f = 20$) of labeled training data (and 10 videos of unlabeled training data used in similarity constraints), our method (**da + lap-sim**) shows a 10.8% improvement over the **da only** baseline of using the PMT-SVM learning technique alone (with no similarity constraints), boosting average precision (AP) from 0.3530 to 0.3913. Additionally, our method shows a 21.5% improvement over using an SVM trained on source

N_f	**dpm**$_S$	**dpm**$_{S \cup T}$	**da only**	**da + lap-sim**
10	0.3220	**0.3502**	0.1121	*0.3317*
20	0.3220	0.3473	*0.3530*	**0.3913**
30	0.3220	0.3816	**0.4306**	**0.4303**
40	0.3220	0.4314	*0.4538*	**0.4584**

Table 5. Detection results (AP) on the *PASCAL 2007* → *VisInt* domain shift. N_f shown is the number of frames in the training set, drawn from $N_f/10$ videos. In the *sim* experiments, $5N_f$ frames are additionally used as unlabeled data, taken from $5N_f/10$ videos. Our results show that using domain adaptation techniques and similarity constraints significantly improves over using either one alone or neither.

data alone (**dpm**$_S$). Figure 4 shows samples of successful detections for our method (right) that were missed by the **da only** baseline (left). In many cases, the baseline detector fires on a tree or other background element, while our model trained with similarity constraints does not.

As the training set size N_f increases, the marginal performance boost provided by the similarity constraints decreases. However, the results for lower N_f clearly demonstrate that similarity constraints can be highly informative in a video detection setting when labeled data is relatively scarce. The setting of domain adaptation to a target domain with a small amount of labeled data is very important, as it allows users to take a generic "off-the-shelf" detector (such as one trained on the PASCAL dataset), label a small amount of data (10 frames in 1 or 2 videos), and use domain adaptation techniques to refine the generic detector for high accuracy on the dataset of interest.

7. Conclusion

We have proposed a new approach for semi-supervised domain adaptation, which explicitly makes use of similarity constraints in the target domain to improve adaptation performance and to enrich learning with unlabeled training examples. Our method is based on manifold regularization and we showed how to extend two different supervised domain adaptation methods, the PMT-SVM from [1], and MMDT, a transform-based approach proposed by [16].

Additionally, we demonstrated the suitability of our method for multi-category classification in multi-view data and object detection in video. These are two applications where similarity constraints can be obtained with low or zero manual annotation cost. For multi-view classification, constraints can be obtained by grouping views of the same object instance, for videos, by grouping examples that are on the same motion trajectory. Finally, we showed how to extend the deformable parts model [12] to semi-supervised domain adaptation.

Our experimental results show that using similarity constraints to incorporate knowledge about unlabeled examples significantly improves recognition performance for both scenarios. In general, our algorithm contributes a new formulation to seamlessly incorporate instance constraints into a wide class of semi-supervised domain adaptation algorithms.

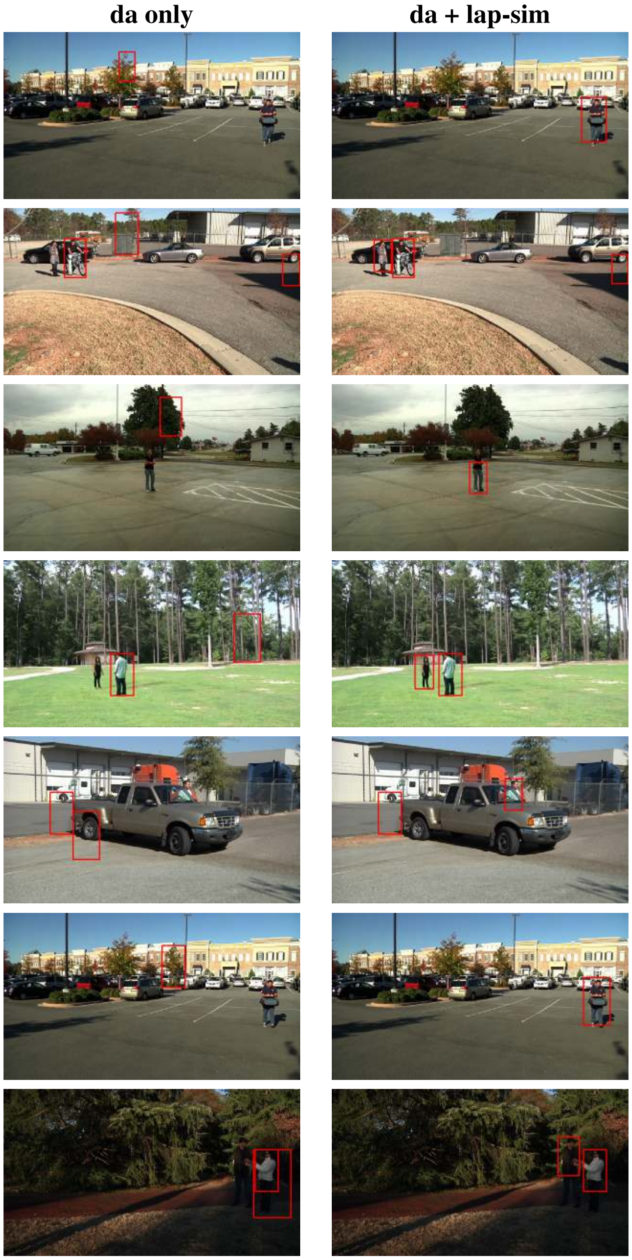
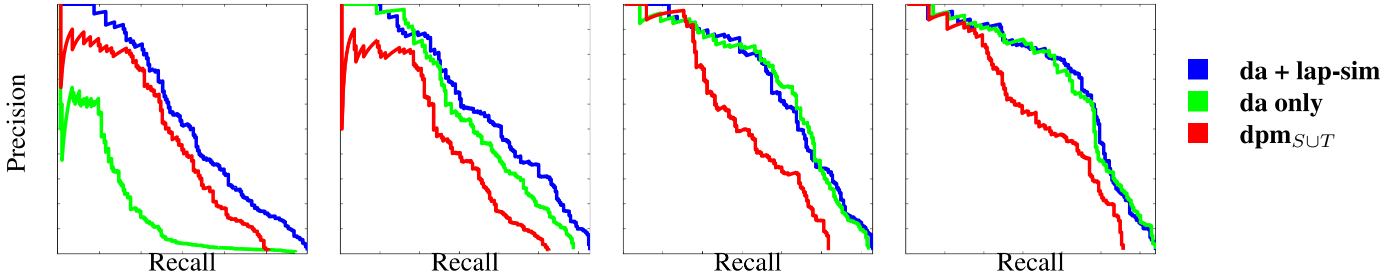


Figure 4. Sample detections from the best performing baseline model (**da only**, left) and from our model (**da + lap-sim**, right) at $N_f = 20$.

For future work, we plan to perform adaptation on the part level of the detector, which includes the appearance of each part as well as the constellation between them. Another research direction will be to extend our methods to



$N_f = 10$ $N_f = 20$ $N_f = 30$ $N_f = 40$

Figure 3. Precision-recall curves for the *PASCAL 2007* → *VisInt* domain shift (best viewed in color).

large-scale scenarios with thousands of unlabeled videos obtained from internet sources.

Acknowledgements: This work was supported by NSF grants IIS-1116411 and IIS-1212798, DARPA, and the Toyota Corporation.

References

[1] Y. Aytar and A. Zisserman. Tabula rasa: Model transfer for object category detection. In *CVPR*, 2011. 1, 2, 3, 6, 7

[2] A. Bergamo and L. Torresani. Exploiting weakly-labeled web images to improve object classification: a domain adaptation approach. In *NIPS*, 2010. 2

[3] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. V. Gool. Robust tracking-by-detection using a detector confidence particle filter. In *ICCV*, 2009. 2

[4] C. Christoudias, R. Urtasun, and T. Darrell. Multi-view learning in the presence of view disagreement. In *UAI*, 2008. 2

[5] W. Dai, Y. Chen, G.-R. Xue, Q. Yang, and Y. Yu. Translated learning: Transfer learning across different feature spaces. In *NIPS*, 2008. 2

[6] H. Daume III. Frustratingly easy domain adaptation. In *ACL*, 2007. 2

[7] L. Duan, I. W. Tsang, D. Xu, and T.-S. Chua. Domain adaptation from multiple sources via auxiliary classifiers. In *ICML*, 2009. 1

[8] L. Duan, I. W. Tsang, D. Xu, and S. J. Maybank. Domain transfer SVM for video concept detection. In *CVPR*, 2009. 2

[9] L. Duan, D. Xu, and I. Tsang. Learning with augmented features for heterogeneous domain adaptation. In *ICML*, 2012. 1, 2

[10] M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes (VOC) challenge. *IJCV*, 88(2):303–338, 2010. 6

[11] A. Farhadi and M. K. Tabrizi. Learning to recognize activities from the wrong view point. In *ECCV*, 2008. 2

[12] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *PAMI*, 32(9):1627–1645, 2010. 3, 4, 6, 7

[13] B. Gong, Y. Shi, F. Sha, and K. Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *CVPR*, 2012. 1, 2, 5

[14] R. Gopalan, R. Li, and R. Chellappa. Domain adaptation for object recognition: An unsupervised approach. In *ICCV*, 2011. 1, 2

[15] D. Hardoon, S. Szedmak, and J. Shawe-Taylor. Canonical correlation analysis: an overview with application to learning methods. *Neural Computation*, 16:26392664, 2004. 2

[16] J. Hoffman, E. Rodner, J. Donahue, T. Darrell, and K. Saenko. Efficient learning of domain-invariant image representations. In *ICLR*, 2013. 1, 2, 3, 4, 7

[17] J. Jiang. A literature survey on domain adaptation of statistical classifiers. http://sifaka.cs.uiuc.edu/jiang4/domain_adaptation/survey/. 2

[18] B. Kulis, K. Saenko, and T. Darrell. What you saw is not what you get: Domain adaptation using asymmetric kernel transforms. In *CVPR*, 2011. 1, 2, 5

[19] Y. J. Lee, J. Kim, and K. Grauman. Key-segments for video object segmentation. In *ICCV*, 2011. 1

[20] X. Li. *Regularized Adaptation: Theory, Algorithms and Applications*. PhD thesis, University of Washington, 2007. 2

[21] B. Long, P. S. Yu, and Z. Zhang. A general model for multiple view unsupervised learning. In *ICDM*, 2009. 2

[22] S. Melacci and M. Belkin. Laplacian support vector machines trained in the primal. *JMLR*, 12:1149–1184, 2011. 2, 3

[23] A. Prest, C. Leistner, J. Civera, C. Schmid, and V. Ferrari. Learning object class detectors from weakly annotated video. In *CVPR*, 2012. 2, 4

[24] K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Adapting visual category models to new domains. In *ECCV*, 2010. 1, 2

[25] K. Saenko, B. Packer, C.-Y. Chen, S. Bandla, Y. Lee, Y. Jia, J.-C. Niebles, D. Koller, L. Fei-Fei, K. Grauman, and T. Darrell. Mid-level features improve recognition of interactive activities. Technical Report UCB/EECS-2012-209, EECS Department, University of California, Berkeley, 2012. 6

[26] Y. Shi and F. Sha. Information-theoretical learning of discriminative clusters for unsupervised domain adaptation. In *ICML*, 2012. 2

[27] J. Yang, R. Yan, and A. Hauptmann. Adapting SVM classifiers to data with shifted distributions. In *ICDM Workshops*, 2007. 2

[28] J. Yang, R. Yan, and A. G. Hauptmann. Cross-domain video concept detection using adaptive SVMs. *ACM Multimedia*, 2007. 1