Heterogeneous Multi-Group Adaptation for Event Recognition in Consumer Videos

Mingyu Yao[1], Xinxiao Wu[1], Mei Chen[2] and Yunde Jia[1]

[1] Beijing Lab of Intelligent Information Technology, School of Computer Science,
Beijing Institute of Technology, Beijing 100081 CHINA
[2] Electrical and Computer Engineering, State University of New York at Albany,
NY 12222, USA
{yaomingyu,wuxinxiao,jiayunde}@bit.edu.cn,meichen@albany.edu

Abstract. Event recognition in consumer videos has attracted much attention from researchers. However, it is a very challenging task since annotating numerous training samples is time consuming and labor expensive. In this paper, we take a large number of loosely labeled Web images and videos represented by different types of features from Google and YouTube as heterogeneous source domains, to conduct event recognition in consumer videos. We propose a heterogeneous multi-group adaptation method to partition loosely labeled Web images and videos into several semantic groups and find the optimal weight for each group. To learn an effective target classifier, a manifold regularization is introduced into the objective function of Support Vector Regression (SVR) with an ϵ-insensitive loss. The objective function is alternatively solved by using standard quadratic programming and SVR solvers. Comprehensive experiments on two real-world datasets demonstrate the effectiveness of our method.

Keywords: Event recognition, Multi-group adaptation, Transferring learning

1 Introduction

Event recognition in consumer videos has been an active field in computer vision because of its multitude of applications in video retrieval and classification. Many existing studies [8, 9, 18] have shown good performances in event analysis by using a large number of labeled training videos to learn a robust classifier. Collecting sufficient videos and annotating them are labor expensive and time consuming tasks. Fortunately, Web search engines have become increasingly mature and can provide abundant loosely labeled data, which is beneficial for researchers to collect loosely labeled training data instead of manual annotation. Several methods [4, 5, 19] have been proposed to adapt the knowledge learned from Web domain (source domain) to consumer domain (target domain). Duan et al. [5] proposed a domain selection machine method for event recognition in consumer videos by exploiting labeled Web images from different sources. Wang et al. [19] proposed a model to annotate the labels of target domain videos with

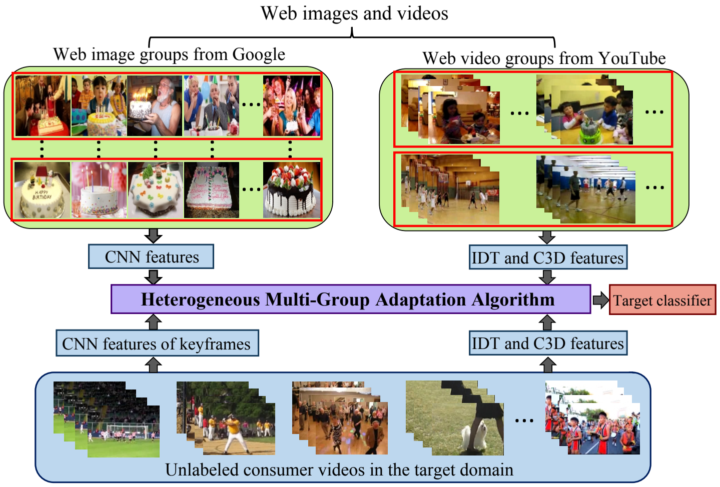
Web images and videos
Web image groups from Google
Web video groups from YouTube
CNN features
IDT and C3D features
Heterogeneous Multi-Group Adaptation Algorithm
Target classifier
CNN features of keyframes
IDT and C3D features
Unlabeled consumer videos in the target domain

Fig. 1. An illustration of the proposed heterogeneous multi-group adaptation framework.

the knowledge learned from a large number of labeled Web images and a few labeled videos. Chen et al. [4] considered temporal information and leveraged a large number of labeled Web images and videos for visual event recognition in consumer videos. However, the above methods only leverage Web images or videos according to their sources and ignore their intrinsic semantic relationships.

In this paper, we divide loosely labeled Web images and videos into several semantic groups to conduct event recognition in consumer videos where there are no labeled consumer videos. In order to comprehensively describe the events in consumer videos, we utilize several query keywords corresponding to specific classes of event in target domain to collect Web images and videos, respectively. For example, we use the keywords “birthday”, “birthday cake”, “birthday party”, “celebration, and “anniversary” to search on the Web for images and videos related to the event “birthday. The returned images and videos of each keyword are regarded as an image group and a video group. Different group samples are represented by different types of features. The Convolutional Neural Networks (CNNs) [16] are used to represent Web image groups, and the Improved Dense Trajectories (IDT) [20] and the Convolutional 3D (C3D) [17] are used to represent Web video groups. All types of features in Web image groups and Web video groups are used to represent the consumer videos (see Fig. 1).

The downloaded images and videos by keyword search are often of poor quality and are not closely related to specific classes of event in the target domain. Therefore directly transferring the knowledge from these images and videos might lead to a negative transfer problem. To address this issue, we have developed a heterogeneous multi-group adaptation method to find the optimal weight for each group. Larger weight means that the group is more relevant to the target domain. For each group, a new classifier is learned by minimizing the distance between the new classifier and the pre-trained group classifier in terms of their weight vectors. We define the target classifier as a linear combination of these new classifiers. We further introduce a manifold regularization together with the target classifier into the objective of support vector regression with an ϵ-insensitive loss, which is based on the smoothness assumption to make two nearby target samples share the similar decision values in a high-density region. Moreover, we develop an alternating optimization algorithm where standard quadratic programming and support vector regression solvers are used to solve the target objective function, and a linear programming solver is used to find the optimal weight for each group.

The contributions of this paper are: (1) Overcome the lack of labeled videos by dividing loosely labeled Web images and videos from heterogeneous source domains into several semantic groups for event recognition in consumer videos. (2) Incorporate a manifold regularization into an ϵ-SVR based objective function to more effectively learn the target classifier.

2 Related Work

Domain adaptation has been applied for a wide variety of applications, such as event recognition [4, 14, 22] and object recognition [12, 13, 15, 21, 23]. Long et al. [13] proposed a transfer joint matching algorithm to learn a feature space to minimize the distance between source domain instances and target domain instances. They [14] proposed a transfer kernel learning method to directly match the distributions between source data and target data in a reproducing kernel Hilbert space. Sener et al. [15] jointly optimize the representation, target label and cross domain transformation by using deep neural networks.

The above methods [13–15] mainly cope with the single source domain setting. When training samples come from different sources, several multiple source domain adaptation methods [4,6,7] are proposed. Duan et al. [6] proposed a domain adaptation machine method to learn a target classifier by using a set of pre-learned classifiers learned from multiple source domains samples to predict the labels of the target samples. Feng et al. [7] proposed a joint weighting scheme based on smoothness assumption, which makes two similar target samples have the similar decision values and the decision values of positive labeled samples are more higher than negative labeled samples. Chen et al. [4] proposed a multi-domain adaptation with heterogeneous sources method to learn an optimal target classifier where the samples from different sources have different types of features and predict the labels of unlabeled target samples based on multiple types of features.

Different from the setting of [4] which leverages Web videos and images according to their sources, we employ the concept of group to search labeled Web images and videos related to specific event classes in consumer videos, considering their intrinsic semantic relationships. Different types of features with different dimensions are used to represent samples from Web image groups and video groups while all types of features are used to represent target domain samples. We propose an $\epsilon-$SVR based objective function, in which a manifold regularization is introduced to enforce the target classifier to be smooth on the consumer videos. The ϵ-SVR can lead to a sparse representation of the target classifier. Our optimal problem can be efficiently solved using standard quadratic programming and SVR solvers, compared with complex cutting plane method used in [4].

3 Event Recognition based on Heterogeneous Multi-Group Adaptation

In this section, we provide a detailed description of our heterogeneous multi-group adaptation method. Following the terminology of domain adaptation, we refer to the loosely labeled Web image and video domains as heterogeneous source domains and the consumer video domain as the target domain, in which there are no labeled consumer videos.

In order to obtain training samples, we adopt keyword searching method to search images and videos from different engines (*e.g.*, Google.com and YouTube.com) with several keywords related to each event class. The returned search result of each keyword is called a group, which is regarded as a source domain. Since the feature distributions of samples from different groups (*e.g.*, from image groups and video groups) change significantly and the data distributions of samples from groups and target domain are also different, we address the problem of heterogeneous multi-group adaptation in this paper. The data from the g-th group for each event class is denoted as $D^g = \left\{(\mathbf{x}_i^g,\ y_i^g)|_{i=1}^{n_g}\right\}$, $g \in \{1, ..., G\}$, where G is the total number of groups and n_g is the total number of samples in the g-th group. Each sample $\mathbf{x}_i^g$ is assigned a label $y_i^g \in \{-1, 1\}$. All the unlabeled videos from the target domain are denoted as $D^T = \{\mathbf{z}_i^T|_{i=1}^{n_T}\}$, where n_T is the total number of target domain videos. Each video is assigned G views (*i.e.*, $\mathbf{z} = (\mathbf{z}^{[1]}, ..., \mathbf{z}^{[G]})$). The g-th view $\mathbf{z}^{[g]}$ is the same view as the g-th group $\mathbf{x}^g$ and they are in the same feature space.

In the following, we use the superscript $'$ to denote the transpose of a vector or matrix. We also define $\mathbf{0}_n$ and $\mathbf{1}_n$ as the $n \times 1$ column vectors of all zeros and all ones, respectively, and define $\mathbf{I}_n$ and $\mathbf{0}_{n\times m}$ as the $n \times n$ identity matrix and $n \times m$ matrix of all zeros. Moreover, we use $\odot$ to denote the element-wise product between two vectors or two matrices.

3.1 Transfer Model

We focus on learning a robust target classifier. Inspired by [4], our target classifier f^T is formulated as

$$f^T(\mathbf{z}) = \sum_{g=1}^{G} d_g \mathbf{w}_g^{'} \varphi(\mathbf{z}^g), \tag{1}$$

where $\mathbf{w}_g$ is the weight vector of the g-th view feature in the target domain, and $\varphi(\cdot)$ is the feature mapping function, which can induce a kernel function in the Reproducing Kernel Hilbert Space (RKHS), $k(\mathbf{x}_i, \mathbf{x}_j) = \varphi^{'}(\mathbf{x}_i)\varphi(\mathbf{x}_j)$. d_g represents the weight of the decision values of the g-th view classifier. Chen et al. [4] expanded the regularization term in [2] as $\sum_{s=1}^{S} d_s \|\mathbf{w}_s - \gamma_s \mathbf{u}_s\|^2$, which can be applied to deal with multiple source domain setting. d_s measures the distance between the s-th view target classifier and the source classifier, and will have a larger value in case the distance is closer.

In this section, we use the expanded regularization term to penalize the complexity of the target classifier $\mathbf{f}^T$, which is minimized in our objective function. A manifold regularization $\mathbf{f}^{T\prime}\mathbf{L}\mathbf{f}^T$ is introduced to keep the target classifier s-mooth on the data manifold, namely, the two nearby patterns in a high-density region should share similar decision values. In order to obtain a sparse representation, we integrate the above two regularization terms into the objective of support vector regression with an $\epsilon-$insensitive loss. We formulate our objective function as

$$\min_{\substack{\mathbf{d},\mathbf{w}_g,\gamma_g,\mathbf{f}^T\\ \xi_i^g,\xi_i^T,\xi_i^{*T}}} \frac{1}{2}(\sum_g d_g \|\mathbf{w}_g - \gamma_g \mathbf{u}_g\|_2^2 + \theta \sum_g \gamma_g^2)$$

$$+ C_T \sum_i l_\varepsilon(f^T(x_i^T) - f_i^T) + C_G \sum_g \sum_i \xi_i^g + \mathbf{f}^{T\prime}\mathbf{L}\mathbf{f}^T, \tag{2}$$

where θ, C_T, C_G are regularization parameters used for balancing different terms, and $\mathbf{d} = [d_1, .., d_g]'$ is the weight vector. $\mathbf{f}^T$ is the decision values of all the target samples, denoted as $\mathbf{f}^T = [f_1^T, ..., f_{n_T}^T]'$, on which the graph Laplacian matrix $\mathbf{L}$ is computed. $\mathbf{L}$ is denoted as $\mathbf{L} = \mathbf{I} - \mathbf{D}^{-0.5}\mathbf{W}\mathbf{D}^{-0.5}$, where $\mathbf{W} = (W_{ij})$ and $\mathbf{D}$ is a diagonal matrix, denoted as $D_{ii} = \sum_{j=1}^{n} W_{ij}$. The l_2 norm of γ_g is used as a penalized term in order to avoid over-fitting. $\sum_i l_\varepsilon(f^T(x_i^T) - f_i^T)$ is the empirical error of the target classifier f^T on all the target data, and l_ε is $\epsilon-$insensitive loss: $l_\varepsilon(t) = \begin{cases} \mid t \mid - \epsilon, & if \mid t \mid > \epsilon; \\ 0, & otherwise. \end{cases}$. Since $\epsilon-$insensitive loss is non-smooth, Eq. (2) is usually transformed into a constrained optimization problem :

$$\min_{\substack{\mathbf{d},\mathbf{w}_g,\gamma_g,\mathbf{f}^T\\ \xi_i^g,\xi_i^T,\xi_i^{*T}}} \frac{1}{2}(\sum_g d_g \|\mathbf{w}_g - \gamma_g \mathbf{u}_g\|_2^2 + \theta \sum_g \gamma_g^2) + C_T \sum_i (\xi_i^T + \xi_i^{*T})$$

$$+ C_G \sum_g \sum_i \xi_i^g + \mathbf{f}^{T\prime}\mathbf{L}\mathbf{f}^T, \tag{3}$$

$$s.t. \quad \sum_{g=1}^{G} d_g \mathbf{w}_g' \boldsymbol{\varphi}_g(\mathbf{z}_i^{[g]}) - f_i^T \leq \varepsilon + \xi_i^T, \ \xi_i^T \geq 0, \tag{4}$$

$$f_i^T - \sum_{g=1}^{G} d_g \mathbf{w}_g' \boldsymbol{\varphi}_g(\mathbf{z}_i^{[g]}) \leq \varepsilon + \xi_i^{*T}, \ \xi_i^{*T} \geq 0, \tag{5}$$

$$d_g y_i^g \mathbf{w}_g' \boldsymbol{\varphi}_g(\mathbf{x}_i^g) \geq 1 - \xi_i^g, \ \xi_i^g \geq 0, \ \mathbf{d} \geq \mathbf{0}, \ \mathbf{1}'\mathbf{d} = 1, \tag{6}$$

where ξ_i^g is the slack variable of the g-th group training samples, ξ_i^T and ξ_i^{*T} are the slack variables of all the target samples.

3.2 Detailed Solution

In order to solve the optimization problem in Eq. (3), we introduce the Lagrangian multipliers α_i^T's, α_i^{*T}'s, α_i^g's, η_i^T's, η_i^{*T}'s and η_i^g's for the constraints in (4), (5) and (6), and then have the following Lagrangian function:

$$\begin{aligned}L =& \frac{1}{2}\sum_{g=1}^{G} d_g \|\mathbf{w}_g - \gamma_g \mathbf{u}_g\|_2^2 + \frac{1}{2}\theta \sum_{g=1}^{G} \gamma_g^2 + C_T \sum_{i=1}^{n_T} (\xi_i^T + \xi_i^{*T}) + C_G \sum_{g=1}^{G}\sum_{i=1}^{n_g} \xi_i^g \\ & + \mathbf{f}^{T\prime}\mathbf{L}\mathbf{f}^T - \sum_{i=1}^{n_T} \alpha_i^T (\varepsilon + \xi_i^T - \sum_{g=1}^{G} d_g \mathbf{w}_g' \boldsymbol{\varphi}_g(\mathbf{z}_i^{[g]}) + f_i^T) - \sum_{i=1}^{n_T} \eta_i^T \xi_i^T \\ & - \sum_{i=1}^{n_T} \alpha_i^{*T} (\varepsilon + \xi_i^{*T} + \sum_{g=1}^{G} d_g \mathbf{w}_g' \boldsymbol{\varphi}_g(z_i^{[g]}) - f_i^T) - \sum_{i=1}^{n_T} \eta_i^{*T} \xi_i^{*T} \\ & - \sum_{g=1}^{G}\sum_{i=1}^{n_g} \alpha_i^g (y_i^g d_g \mathbf{w}_g' \boldsymbol{\varphi}_g(x_i^g) - 1 + \xi_i^g) - \sum_{i=1}^{G}\sum_{i=1}^{n_g} \eta_i^g \xi_i^g .\end{aligned} \tag{7}$$

By taking the derivatives of the Lagrangian function with respect to variables $w_g, \gamma_g, f^T, \xi_i^g, \xi_i^T, \xi_i^{*T}$ to zeros, Eq. (3) can be converted into the following dual form:

$$\min_{\mathbf{d}} \max_{\boldsymbol{\alpha^T}\boldsymbol{\alpha^{*T}}\boldsymbol{\alpha}_g} -\frac{1}{2}\boldsymbol{\alpha}'(\sum_g d_g \tilde{\mathbf{K}}^{[g]} \odot \mathbf{y}\mathbf{y}')\boldsymbol{\alpha} - \varepsilon \mathbf{1}'(\boldsymbol{\alpha^T} + \boldsymbol{\alpha^{*T}}) + \sum_g \mathbf{1}'\boldsymbol{\alpha}_g, \tag{8}$$

where $\boldsymbol{\alpha} = [\boldsymbol{\alpha}_1', ..., \boldsymbol{\alpha}_G', (\boldsymbol{\alpha}^T - \boldsymbol{\alpha}^{*T})']'$, and $\boldsymbol{\alpha}_g = [\alpha_1^g, ..., \alpha_{n_g}^g]'$, $\mathbf{0} \le \boldsymbol{\alpha}_g \le C_G\mathbf{1}$, $g = 1, ..., G$, $-C_T\mathbf{1} \le \boldsymbol{\alpha}^T, \boldsymbol{\alpha}^{*T} \le C_T\mathbf{1}$. The labels of training samples are defined as $\mathbf{y} = [\mathbf{y}_1', ..., \mathbf{y}_G', \mathbf{1}_{n_T'}]'$, where $\mathbf{y}_g = [y_1^g, ..., y_{n_g}^g]'$ represents the labels of the g-th group samples. $\tilde{\mathbf{K}}^{[g]}$ is the newly transformed kernel matrix computed on all the samples from the g-th group and the g-th view of target domain, which is defined as

$$\tilde{\mathbf{K}}^{[g]} = \mathbf{K}^{[g]} + \frac{1}{\theta}\mathbf{f}^{[g]}\mathbf{f}^{[g]\prime} + \frac{1}{2}\begin{bmatrix}\mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{L}^{-1}\end{bmatrix}, \tag{9}$$

where $\mathbf{K}^{[g]} = \boldsymbol{\varphi}^{[g]\prime}\boldsymbol{\varphi}^{[g]}$, and $\boldsymbol{\varphi}^{[g]} = [\mathbf{0}_{h_g \times N(1,g-1)}, \boldsymbol{\varphi}_g, \mathbf{0}_{h_g \times N(g+1,G)}, \boldsymbol{\varphi}_T^{[g]}]$. $\boldsymbol{\varphi}_g = [\boldsymbol{\varphi}_g(\mathbf{x}_1^g), ..., \boldsymbol{\varphi}_g(\mathbf{x}_{n_g}^g)]$ and $\boldsymbol{\varphi}_T^{[g]} = [\boldsymbol{\varphi}_g(\mathbf{z}_1^{[g]}), ..., \boldsymbol{\varphi}_g(\mathbf{z}_{n_T}^{[g]})]$ are defined as the mapped feature matrices by nonlinear mapping on the samples of the g-th group and the g-th view of target domain respectively. The dimension of $\boldsymbol{\varphi}_g(\mathbf{x}^g)$ is h_g, $\boldsymbol{\varphi}_g \in R^{h_g \times n_g}$ and $\boldsymbol{\varphi}_T^{[g]} \in R^{h_g \times n_T}$. $\mathbf{f}^{[g]}$ is the decision values of $\boldsymbol{\varphi}^{[g]}$ using the g-th group classifier $f^g(\mathbf{x})$, defined as $\mathbf{f}^{[g]} = [\mathbf{0}_{N(1,g-1)}', \mathbf{f}_g', \mathbf{0}_{N(g+1,G)}', \mathbf{f}_T^{[g]\prime}]$, where $\mathbf{f}_g = [f^g(\mathbf{x}_1^g), ..., f^g(\mathbf{x}_{n_g}^g)]'$ and $\mathbf{f}_T^{[g]} = [f^g(\mathbf{z}_1^{[g]}), ..., f^g(\mathbf{z}_{n_T}^{[g]})]'$ are decision values of the g-th group and the g-th view of target domain. N(m,n) is the total number of samples between the m-th group and the n-th group.

$\tilde{\mathbf{K}}^{[g]} \odot \mathbf{y}\mathbf{y}'$ can be rewritten as

$$\tilde{\mathbf{K}}^{[g]} \odot \mathbf{y}\mathbf{y}' = \begin{bmatrix}\mathbf{K}_{GG}^{[g]} & \mathbf{K}_{GT}^{[g]} \\ \mathbf{K}_{TG}^{[g]} & \mathbf{K}_{TG}^{[g]}\end{bmatrix}, \tag{10}$$

where $\mathbf{K}^{[g]}_{GG} \in R^{n_g \times n_g}$, $\mathbf{K}^{[g]}_{GT} \in R^{n_g \times n_T}$, $\mathbf{K}^{[g]}_{TG} \in R^{n_T \times n_g}$, and $\mathbf{K}^{[g]}_{TT} \in R^{n_T \times n_T}$ are kernel matrices of the g-th group samples and the g-th view of target domain samples. Substituting Eq. (10) into Eq. (8), Eq. (8) can be transformed into the following form:

$$\begin{aligned}\min_{\mathbf{d}} \max_{\boldsymbol{\alpha^T}\boldsymbol{\alpha^{*T}}\boldsymbol{\alpha}_g} \; & -\frac{1}{2}\boldsymbol{\alpha}^{G'}(\sum_g d_g \mathbf{K}^{[g]}_{GG})\boldsymbol{\alpha}^G \\ & + \mathbf{1}'\boldsymbol{\alpha}^G - (\boldsymbol{\alpha}^T - \boldsymbol{\alpha}^{*T})'(\sum_g d_g \mathbf{K}^{[g]}_{TG})\boldsymbol{\alpha}^G \qquad (11) \\ & -\frac{1}{2}(\boldsymbol{\alpha^T} - \boldsymbol{\alpha^{*T}})'(\sum_g d_g \mathbf{K}^{[g]}_{TG})(\boldsymbol{\alpha^T} - \boldsymbol{\alpha^{*T}}) - \varepsilon\mathbf{1}'(\boldsymbol{\alpha^T} + \boldsymbol{\alpha^{*T}}),\end{aligned}$$

where $\boldsymbol{\alpha}^G = [\boldsymbol{\alpha}'_1, ..., \boldsymbol{\alpha}'_G]'$ represents the Lagrangian multipliers of all group samples.

3.3 Optimization Algorithm

In this section, we develop an alternating optimization algorithm to optimize the weight coefficient $\mathbf{d}$, dual variable $\boldsymbol{\alpha}^T$ and $\boldsymbol{\alpha}^{*T}$, and the variable $\boldsymbol{\alpha}^G$ since Eq. (11) is proved to be convergent. The optimization algorithm is divided into two steps.

Update $\boldsymbol{\alpha}^T$, $\boldsymbol{\alpha}^{*T}$ and $\boldsymbol{\alpha}^G$. Alternatively updating $\boldsymbol{\alpha}^T$, $\boldsymbol{\alpha}^{*T}$ and $\boldsymbol{\alpha}^G$ after $\mathbf{d}$ is fixed, we find Eq. (11) can be optimized via two processes. When $\boldsymbol{\alpha}^G$ is fixed, Eq. (11) has the similar form with the standard ϵ-SVR, and we can use the existing toolkit such as LIBSVM [3] to solve $\boldsymbol{\alpha}^T$ and $\boldsymbol{\alpha}^{*T}$. Once $\boldsymbol{\alpha}^T$ and $\boldsymbol{\alpha}^{*T}$ are solved, $\boldsymbol{\alpha}^G$ can be computed by using a standard quadratic programming solver. The updating processes are stopped when Eq. (11) converges to a certain value.

Update d. After solving $\boldsymbol{\alpha}^T$, $\boldsymbol{\alpha}^{*T}$ and $\boldsymbol{\alpha}^G$ in each iteration of the first step, $\mathbf{d}$ can be updated with a linear programming solver.

Algorithm 1: Heterogeneous Multi-Group Adaptation

```
Initialize $\mathbf{d}^o$, which satisfies $D = \{\mathbf{d} \mid \mathbf{d} \geq \mathbf{0}, \mathbf{1}'\mathbf{d} = 1\}, \mathbf{d}^o = \mathbf{d}^O$
repeat
    Substitute $\mathbf{d}^o$ into Eq. (11), update $\boldsymbol{\alpha}^T, \boldsymbol{\alpha}^{*T}, \boldsymbol{\alpha}^G$
    repeat
        Initialize $\boldsymbol{\alpha}^G$, obtain optimal $\boldsymbol{\alpha}^T, \boldsymbol{\alpha}^{*T}$ by using LIBSVM [3]
        Fixing $\boldsymbol{\alpha}^T, \boldsymbol{\alpha}^{*T}$, update $\boldsymbol{\alpha}^G$ using quadratic programming solver
    until The objective of Eq. (11) converges;
    Substitute $\boldsymbol{\alpha}^T$, $\boldsymbol{\alpha}^{*T}$ and $\boldsymbol{\alpha}^G$ into Eq. (11), update $\mathbf{d}$ using linear
    programming solver
    $\mathbf{d}^{o+1} \leftarrow \mathbf{d}^o$
until The objective of Eq. (11) converges;
```

Alternating above two steps, Eq. (11) can quickly converge to a minimum value. Specifically, the optimization details are summarized in Algorithm 1.

Substituting $\boldsymbol{\alpha}^T$, $\boldsymbol{\alpha}^{*T}$, $\boldsymbol{\alpha}^G$ and $\mathbf{d}$ into Eq. (1), the target classifier can be rewritten as

$$f^T(\mathbf{z}) = \sum_g d_g \boldsymbol{\beta}'_g (\boldsymbol{\varphi}^{[g]'} \varphi_g(\mathbf{z}^{[g]}) + \frac{d_g}{\theta} \mathbf{f}^{[g]} f^g(\mathbf{z}^{[g]})), \tag{12}$$

where $\boldsymbol{\beta}_g = [(\boldsymbol{\alpha}_1 \odot \mathbf{y}_1)', ..., (\boldsymbol{\alpha}_G \odot \mathbf{y}_G)', (\boldsymbol{\alpha}^{*T} - \boldsymbol{\alpha}^T)']'$.

4 Experiment

We compare our method with the baseline method employing SVM, the existing single source domain adaptation methods Transfer Joint Matching (TJM) [13] and Transfer Kernel Learning (TKL) [14], and the existing multiple source domain adaptation methods Domain Selection Machine (DSM) [5], Domain Adaptation Machine (DAM) [6], and Multi-domain Adaptation with Heterogeneous Sources (MDA-HS) [4]. All the methods are evaluated on two challenging video datasets CCV [11] and TRECVID MED 2014 [1]. We first introduce the datasets, and then describe the experimental settings. After that, we use Average Precision (AP) to evaluate the performance of all the methods and report the Mean Average Precision (MAP) for all the event classes.

4.1 Datasets and Features

CCV dataset: The CCV dataset [11] contains a training set of 4659 videos and a test set of 4658 videos annotated with 20 semantic categories. Our work focuses on event recognition, therefore, we omit videos from non-event categories (*i.e.*, “beach”, “bird”, “cat”, “dog”, and “playground”). Following [5], we merge “wedding ceremony”, “wedding reception, and “wedding dance” into “wedding”, “non-music performance” and “music performance” into “show”, and “baseball”, “basketball”, “biking”, “ice skating”, “skiing”, “soccer”, “swimming” into “sports”. That yields 2594 videos from 5 event classes (*i.e.*, “birthday”, “parade”, “show”, “sports, and “wedding”).

TRECVID MED 2014 dataset: This dataset consists of 20 event classes (*i.e.*, E021-E040) and a background class, where there are around 100 videos of each event class and 4,983 videos of the background class. In our experiment, we only choose the videos from the event classes as our consumer videos.

Web Image and Video Datasets: In order to construct heterogenous multiple groups, images and videos based on keyword searching are collected from Google and YouTube search engines respectively. We downloaded the top 100 retrieved images and videos for each keyword, and ignore the invalid URLs. This gives us 2547 images and videos for the CCV dataset and 10920 images and videos for the MED dataset.

Features: We extract the CNN features for each image downloaded from Google using Caffe [10] and the VGG model provided by [16]. The output of the second fully-connected layer from the VGG network [16], a 4096-dimensional feature, is extracted as our feature descriptors. For each video in the CCV and MED datasets, we sample 5 frames per second and extract the frame level CNN descriptors. Finally, we apply average pooling on the frame descriptors of each video to generate the video level representation.

For each video in the CCV dataset and downloaded from YouTube, three types of local descriptors (*i.e.*, Histogram of Oriented Gradient, Histogram of Optical Flow and Motion Boundary Histogram) are extracted by using the source codes provided in [20]. The sampling stride is set as 16 while the other parameters are set as default values. Then, following [20], we use Fisher vector to encode these local descriptors and apply l_2 normalization. Finally, the normalized Fisher vectors of different descriptors are concatenated to generate the final video representation. For each video in the MED dataset and downloaded from YouTube, we split it into 16-frame long clips and pass these clips to the C3D network provided by [17] to extract C3D features. The final feature for each video is computed by averaging the clip features followed by an l_2 normalization.

4.2 Experimental Setup

In the experiments, we search five groups from Google and YouTube for each event class. The CCV and MED datasets are used as our target domains. We first pre-train a classifier $f^g(\mathbf{x}^g)$ for each event class on one group, where the positive samples consist of samples belonging to the corresponding event class in the corresponding group, and the negative samples are from all the samples belonging to groups of other event classes. The classifier f^g pre-trained on the g-th group is used to compute the decision values of samples from the g-th view of target domain, and the decision values of target samples are computed by averaging these decision values from all the group classifiers.

The baseline SVM is referred as SVM_A in which the decision values of target samples are computed by fusing the decision values from all the pre-trained classifiers. Since TJM and TKL are single domain adaptation methods, we extend both TJM and TKL into multi-group versions applying the same fusion strategy as SVM_A, denoted as Multi-TJM and Multi-TKL. The experiment settings of multiple source domain adaptation methods (*i.e.*, DAM, DSM, MDA-HS) are different from our experiments, therefore, we regard each group as a source domain. For DAM and DSM, we need to compute decision values of samples from each view of target domain, which are averaged as the final decision values of target samples. For all the methods, we use one-vs-all SVMs with the Gaussian kernel $K(\mathbf{x}_i, \mathbf{x}_j) = exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2)$, where γ is set as the mean of the squared distance between all the training samples. The regularization parameters C_G and C_T are set as 1 and 10 respectively since target samples is more important for classification. θ and ϵ are empirically set to 0.002 and 10^{-5}, respectively.

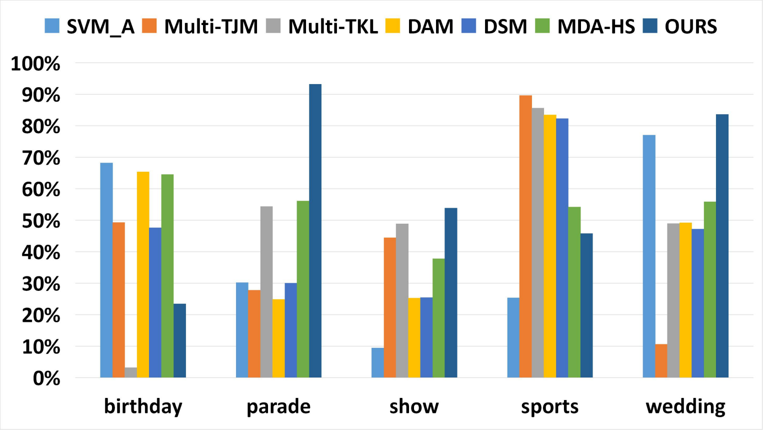

Fig. 2. Per-event APs on the CCV datasets.

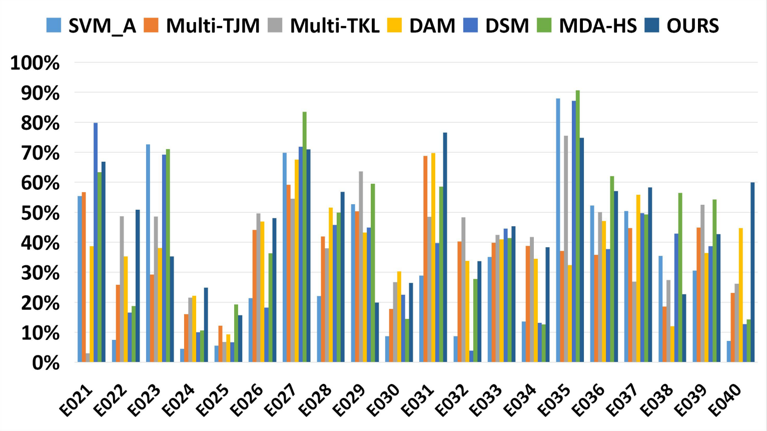

Fig. 3. Per-event APs on the MED dataset.

4.3 Results

For all the methods, the per-event APs on the CCV and MED datasets are plotted in Fig. 2 and Fig. 3, and the MAPs on both datasets are showed in Tables 1. It is interesting to observe that our method outperforms other methods on both datasets, which demonstrates the effectiveness of our proposed ϵ-SVR based objective function and introduced manifold regularization. There is no consistent winner among DSM, DAM and Multi-TKL in terms of MAPs, which indicates that event recognition in consumer videos from heterogeneous groups is a challenging task. SVM_A is worse than other methods, which demonstrates that adapting the knowledge learned from source domains is beneficial to learn a better classifier in target domain. MDA-HS and our method achieve relatively

Table 1. MAP (%) of all methods on the CCV and MED datasets.

Method	SVM_A	Multi-TJM	Multi-TKL	DAM	DSM	MDA-HS	OURS
CCV	42.12	44.40	48.25	49.68	46.58	53.74	60.78
MED	33.51	37.26	40.02	39.53	37.78	44.54	46.26

high performances since both methods can deal with heterogeneous features better than other methods. Our method is superior to MDA-HS in terms of computational complexity. Our method solves the objective function by using standard quadratic programming and SVR solvers. In MDA-HS, cutting-plane algorithm and multiple kernel learning are applied to solve the optimization problem, which are computationally intensive.

5 Conclusion and Future Work

In this paper, we have proposed a heterogeneous multi-group adaptation method to leverage a large number of loosely labeled Web images (from Google) and videos (from YouTube) to recognize complex events in target domain where there are no labeled videos. These images and videos are divided into several semantic groups based on different keywords corresponding to specific classes of event in the target domain. Our method can learn the weights of the classifiers from corresponding view of target domain as well as the weights of classifier learned from different groups. A manifold regularization is introduced to enforce the target classifier to be smooth on target samples.

For future work, we plan to experiment with using deep neural networks to learn discriminative deep representations for both source and target data with the intent to improve event recognition in consumer videos.

Acknowledgments. This work was supported in part by the Natural Science Foundation of China (NSFC) under Grants No. 61673062 and No. 61472038.

References

1. Trecvid med 14. http://www.nist.gov/itl/iad/mig/med14.cfm.
2. Aytar, Y., Zisserman, A.: Tabula rasa: Model transfer for object category detection. In: IEEE International Conference on Computer Vision. pp. 2252–2259. IEEE (2011)
3. Chang, C.C., Lin, C.J.: Libsvm: a library for support vector machines. ACM Transactions on Intelligent Systems and Technology 2(3), 27 (2011)
4. Chen, L., Duan, L., Xu, D.: Event recognition in videos by learning from heterogeneous web sources. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2666–2673 (2013)
5. Duan, L., Xu, D., Chang, S.F.: Exploiting web images for event recognition in consumer videos: A multiple source domain adaptation approach. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 1338–1345. IEEE (2012)

6. Duan, L., Xu, D., Tsang, I.W.H.: Domain adaptation from multiple sources: A domain-dependent regularization approach. IEEE Transactions on Neural Networks and Learning Systems 23(3), 504–518 (2012)
7. Feng, Y., Wu, X., Wang, H., Liu, J.: Multi-group adaptation for event recognition from videos. In: 22nd International Conference on Pattern Recognition. pp. 3915–3920. IEEE (2014)
8. Ikizler-Cinbis, N., Sclaroff, S.: Object, scene and actions: Combining multiple features for human action recognition. In: European Conference on Computer Vision. pp. 494–507. Springer (2010)
9. Izadinia, H., Shah, M.: Recognizing complex events using large margin joint low-level event model. In: European Conference on Computer Vision. pp. 430–444. Springer (2012)
10. Jia, Y.: An open source convolutional architecture for fast feature embedding (2013)
11. Jiang, Y.G., Ye, G., Chang, S.F., Ellis, D., Loui, A.C.: Consumer video understanding: A benchmark database and an evaluation of human and machine performance. In: Proceedings of the 1st ACM International Conference on Multimedia Retrieval. p. 29. ACM (2011)
12. Long, M., Wang, J., Cao, Y., Sun, J., Philip, S.Y.: Deep learning of transferable representation for scalable domain adaptation. IEEE Transactions on Knowledge and Data Engineering 28(8), 2027–2040 (2016)
13. Long, M., Wang, J., Ding, G., Sun, J., Yu, P.S.: Transfer joint matching for unsupervised domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1410–1417 (2014)
14. Long, M., Wang, J., Sun, J., Philip, S.Y.: Domain invariant transfer kernel learning. IEEE Transactions on Knowledge and Data Engineering 27(6), 1519–1532 (2015)
15. Sener, O., Song, H.O., Saxena, A., Savarese, S.: Learning transferrable representations for unsupervised domain adaptation. In: Advances in Neural Information Processing Systems. pp. 2110–2118 (2016)
16. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
17. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3d convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4489–4497 (2015)
18. Vahdat, A., Cannons, K., Mori, G., Oh, S., Kim, I.: Compositional models for video event detection: A multiple kernel learning latent variable approach. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1185–1192 (2013)
19. Wang, H., Wu, X., Jia, Y.: Annotating videos from the web images. In: 21st International Conference on Pattern Recognition. pp. 2801–2804. IEEE (2012)
20. Wang, H., Schmid, C.: Action recognition with improved trajectories. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 3551–3558 (2013)
21. Xu, Y., Fang, X., Wu, J., Li, X., Zhang, D.: Discriminative transfer subspace learning via low-rank and sparse representation. IEEE Transactions on Image Processing 25(2), 850–863 (2016)
22. Yang, X., Zhang, T., Xu, C.: Cross-domain feature learning in multimedia. IEEE Transactions on Multimedia 17(1), 64–78 (2015)
23. Yao, T., Pan, Y., Ngo, C.W., Li, H., Mei, T.: Semi-supervised domain adaptation with subspace learning for visual recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2142–2150 (2015)