Enhancing Action Recognition by Cross-Domain Dictionary Learning

Fan Zhu
fan.zhu@sheffield.ac.uk
Ling Shao
ling.shao@sheffield.ac.uk

Department of Electronic and Electrical Engineering
The University of Sheffield
Sheffield, S1 3JD, UK

Abstract

We present a novel cross-dataset action recognition framework that utilizes relevant actions from other visual domains as auxiliary knowledge for enhancing the learning system in the target domain. The data distribution of relevant actions from a source dataset is adapted to match the data distribution of actions in the target dataset via a cross-domain discriminative dictionary learning method, through which a reconstructive, discriminative and domain-adaptive dictionary-pair can be learned. Using selected categories from the HMDB51 dataset as the source domain actions, the proposed framework achieves outstanding performance on the UCF YouTube dataset.

1 Introduction

In real-world applications, due to the high price of human manual annotation and environmental restrictions, sufficient training data that stay in the same feature space or share the same distribution with the testing data cannot always be guaranteed, in which case insufficient training data can limit the potential discriminability of the trained model. Typical examples can be found in [3], [8], [19], where only one action template is provided for each action class for training, and [15], where training samples are captured from a different viewpoint. In these situations, obtaining more labeled data is either impossible or expensive, while seeking for an alternative way of using data from other domains as compensation can be seen as a possible and economic solution.

Our work is inspired by two facts of the human vision system. The first fact is that humans are able to learn tens of thousands of visual categories in their life, which leads to the hypothesis that humans achieve such a capability by accumulated information and knowledge [7]. Another fact is that human's visual impressions towards the same action or the same object comes from a wide range, e.g., an action seen from 2D static images *vs*. the same action seen from 3D dynamic movies or an object seen from real-world scenes *vs*. the same object seen from low-resolution online images. These facts can be explained in the computer vision language as the human vision system possesses the ability of spanning the intra-class diversity of the original training instances through transferring prior knowledge. Motivated by the above two facts, we introduce a new action recognition framework that utilizes relevant actions from other domains as auxiliary knowledge (motivated by the first fact) to span the intra-class diversity of the original learning system (motivated by the second fact). In

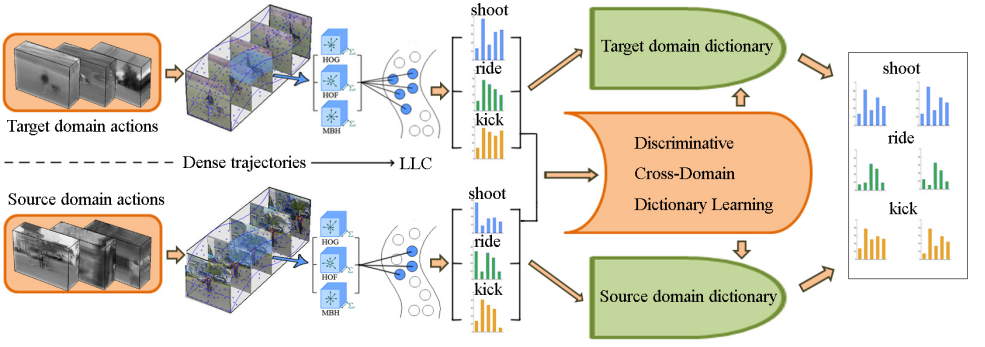

Figure 1: Flowchart of the proposed cross-domain action recognition framework. Local dense trajectory features are extracted from both the target domain actions and the source domain actions, followed by which the local features are coded by LLC. Through the proposed discriminative cross-domain dictionary learning technique, a dictionary pair for both domains is learned from the middle-level action representations, so that actions of the same category from different domains possess similar representations after being coded by the learned dictionary pair.

addition to manually annotated actions in the target domain, labeled actions from a different domain are provided as the source domain actions. Based on the recent success of dictionary learning methods in solving computer vision problems, we present a discriminative cross-domain dictionary learning (DCDDL) technique to learn a reconstructive, discriminative and domain-adaptive dictionary pair for data under different distributions. The flowchart of the proposed framework is shown in Figure 1.

Transfer learning (a.k.a., cross-domain learning, domain transfer, domain adaptation) approaches begin to attract increasing interests in the computer vision community in recent years due to the data explosion on the Internet and the growing demands for visual computation tasks. Domain transfer is used to address the problem of video concept detection in [25] and [5]. The former one utilized the Adaptive Support Vector Machines (A-SVMs) to adapt one or more existing classifiers of any type to a new dataset, and the latter proposed a Domain Transfer Multiple Kernel Learning (DTMKL) method to simultaneously learn a kernel function and a robust SVM classifier by minimizing both the structural risk function of SVM and the distribution mismatch of labeled and unlabeled data in different domains. Duan *et al*. [6] considered to leverage large amounts of loosely labeled web videos for visual event recognition using the Adaptive Multiple Kernel Learning (A-MKL) to fuse the information from multiple pyramid levels of features and cope with the considerable variation in feature distributions between videos across two domains.

Recently, dictionary learning for sparse representation has attracted much attention. It has been successfully applied to a variety of computer vision tasks, e.g., face recognition [24] and image denoising [31]. Using an over-complete dictionary, sparse modeling of signals can approximate the input signal by a sparse linear combination of items from the dictionary. Many algorithms [13], [23], [24] have been proposed to learn such a dictionary according to different criteria. The K-Singular Value Decomposition (K-SVD) algorithm [1] is a classical dictionary learning algorithm that generalizes the K-means clustering pro-

cess for adapting dictionaries to efficiently learn an over-complete dictionary from a set of training signals. The K-SVD method focuses on the reconstructive ability, however, since the learning process is unsupervised, the discriminative capability is not taken into consideration. Consequently, methods that incorporate the discriminative criteria into dictionary learning were proposed in [28], [26], [18], [17], [13], [9]. In addition to the discriminative capability of the learned dictionary, other criteria designed on top of the prototype dictionary learning objective function include multiple dictionary learning [29], category-specific dictionary learning [27], etc. Different from most dictionary learning methods, which learn the dictionary and the classifier separately, the authors of [28] and [10] unified these two learning procedures into a single supervised optimization problem and learned a discriminative dictionary and the corresponding classifier simultaneously. Taking a step further, Qiu et al. [20] and Zheng et al. [30] designed dictionaries for the situations that the present training instances are different from the testing instances. The former presented a general joint optimization function that transforms a dictionary learned from one domain to the other, and applied such a framework to applications such as pose alignment, pose and illumination estimation and face recognition. The latter achieved promising results on the cross-view action recognition problem with pairwise dictionaries constructed using correspondences between the target view and the source view. To make use of some data that may not be relevant to the target domain data, Raina et al. [21] proposed a method that applies sparse coding to unlabeled data to break the tremendous amount of data in the source domain into basic patterns (e.g., edges in the task of image classification) so that knowledge can be transferred through the bottom level to a high level representation.

Our approach differs from the above approaches in such aspects that it more comprehensively learns pairwise dictionaries and a classifier while considering the capacity of the dictionaries in terms of reconstructability, discriminability and domain adaptability. Additionally, corresponding observations across the domains are not required in our framework. Most previous knowledge transfer algorithm focus on the situations where the target domain is incomplete, but have not attempted to utilize other domain data as an aide for enhancing present categorization systems, in our approach, the learned classifier in the target domain becomes more discriminative against intra-class variations as a result of the learning process that integrates with source domain data. Our work makes the following contributions:

⋆ We present a novel cross-domain action recognition framework that attempts to enhance the performance of the original recognition system by spanning the intra-class diversities of the target domain training actions using actions from the source domain.

⋆ The proposed discriminative cross-domain dictionary learning technique copes with the feature distribution mismatch problem across different domains by learning a domain-adaptive dictionary pair that transfers data under different distributions into the same feature space.

⋆ Our approach does not require correspondence annotations across different domains, so that it can be adapted to solve many real-world transfer learning problems.

The remainder of this paper is organized in the following way. In Section 2, the proposed knowledge transfer technique is described in detail including dictionary learning, discriminative cross-domain dictionary learning, optimization and classification. Experimental results on the human action recognition task and performance comparisons with state-of-the-art methods are demonstrated in Section 3. Finally, the conclusion of this work is given in Section 4.

2 Knowledge transfer via discriminative dictionary learning

2.1 Dictionary learning

Let Y_t be the set of target domain n-dimensional input signals, which contain N training instances, i.e., $Y_t = [y_t^1, y_t^2, ..., y_t^N] \in \Re^{n \times N}$. Learning a reconstructive dictionary for obtaining the sparse representation of the target domain signals Y_t can be accomplished by solving the following optimization problem:

$$\begin{aligned} \langle D_t, X_t \rangle = & arg \min_{D_t, X_t} \|Y_t - D_t X_t\|_2^2 \\ & s.t. \forall i, \|x_t^i\|_0 \leq T, \end{aligned} \tag{1}$$

where $D_t = [d_t^1, ...d_t^{K_t}] \in \Re^{n \times K_t}$ denotes the target domain dictionary and $X_t = [x_t^1, ...x_t^N] \in \Re^{K_t \times N}$ denotes the set of sparse signals. The number of dictionary items K_t is set to significantly exceed the number of training instances N to ensure that the dictionary is overcomplete. T is the sparsity constraint factor that limits the number of non-zero elements in the sparse codes, so that the number of items in the decomposition of each signal x_i is less than T.

The choice of a method for dictionary learning critically determines the performance of sparse representation. The K-SVD algorithm [1] is a popular and efficient dictionary learning method that focuses on minimizing the reconstruction error. Some discriminative approaches [28], [26], [18], [17], [16], [2] show their privilege over the K-SVD algorithm by incorporating extra discriminative terms into the objective function for dictionary learning. However, the discriminative terms appear to be introduced to these approaches without considering the data distribution of the training samples, i.e., samples with high confidence possess the same impact as those with low confidence. Such weakness becomes even more severe when dealing with data mismatch scenarios. When allocated with discriminative elements under no smoothness guarantee, performing dictionary learning on both target domain data and mismatched data from a different feature domain can break the smoothness property of the original target domain.

2.2 Cross-domain discriminative dictionary learning

In the source domain, the optimization problem for dictionary learning becomes:

$$\begin{aligned} \langle D_s, X_s \rangle = & arg \min_{D_s, X_s} \|Y_s - D_s X_s\|_2^2 \\ & s.t. \forall i, \|x_s^i\|_0 \leq T, \end{aligned} \tag{2}$$

where $D_s = [d_s^1, ...d_s^{K_s}] \in \Re^{n \times K_s}$ denotes the source domain dictionary and $X_s = [x_s^1, ...x_s^N] \in \Re^{K_s \times N}$ denotes the set of sparse signals. By minimizing the reconstruction error terms $\|Y_t - D_t X_t\|_2^2$ and $\|Y_s - D_s X_s\|_2^2$ in Equation (1) and Equation (2) separately, the sparse representations X_t and X_s still obey to the original distributions in each respective domain. In order to force the mismatched sparse representations from different domains into the same feature space, we combine the objective functions in Equation (1) and Equation (2) in a unified optimization manner and add extra terms to guarantee the overall smoothness in the new

feature space:

$$\begin{aligned}\langle D_t, D_s, X_t, X_s\rangle = arg \min_{D_t,D_s,X_t,X_s} \|Y_t - D_t X_t\|_2^2 \\ +\|Y_s - D_s X_s\|_2^2 + \|X_t - f(Y_t,Y_s)X_s\|_F^2 \\ +\|X_s - f(Y_s,Y_t)X_t\|_F^2 \\ s.t. \forall i, [\ \|x_t^i\|_0, \|x_s^i\|_0\ ] \leq T,\end{aligned} \tag{3}$$

where the function $f(\cdot)$ computes the mapping of correspondence samples (i.e., samples that share the same class labels while being close to each other) across different domains. Thus, small values of $\|X_t - f(Y_t,Y_s)X_s\|_F^2$ and $\|X_s - f(Y_s,Y_t)X_t\|_F^2$ indicate that those data points close to each other are more likely to share the same class label in the new target feature domain and the new source feature domain respectively. Since we are only concerned with the smoothness within the target domain data, the last term in Equation (3) can be removed. According to the stated scenario, no manually annotated correspondences between the target domain data and the source domain data are available in the training phase, thus $f(\cdot)$ is computed using a category-specific searching method. Assuming both Y_t and Y_s are arranged according to their category labels, we can set c_t^1 and c_s^1 as the numbers of the last samples in category 1 in the target domain and the source domain respectively, and similarly c_t^2 and c_s^2 for category 2. Let $\mathbb{A}_1$ be the transition matrix for category 1 and $\mathbb{A}_2$ be the transition matrix for category 2, $\mathbb{A}_1$ and $\mathbb{A}_2$ can then be represented by:

$$\mathbb{A}_1 = \begin{pmatrix} \Psi(y_t^1, y_s^1) & \cdots & \cdots & \Psi(y_t^1, y_s^{c_s^1}) \\ \vdots & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ \Psi(y_t^{c_t^1}, y_s^1) & \cdots & \cdots & \Psi(y_t^{c_t^1}, y_s^{c_s^1}) \end{pmatrix}, \tag{4}$$

$$\mathbb{A}_2 = \begin{pmatrix} \Psi(y_t^{c_t^1+1}, y_s^{c_s^1+1}) & \cdots & \cdots & \Psi(y_t^{c_t^1+1}, y_s^{c_s^2}) \\ \vdots & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ \Psi(y_t^{c_t^2}, y_s^{c_s^1+1}) & \cdots & \cdots & \Psi(y_t^{c_t^2}, y_s^{c_s^2}) \end{pmatrix}, \tag{5}$$

where $\Psi(y_t^i, y_s^j)$ in each $\mathbb{A}_c$ can be computed by the Gaussian kernel:

$$\Psi(y_t^i, y_s^j) = \frac{1}{\sqrt{2\pi}} e^{(-\frac{y_t^{i\,2} - y_s^{j\,2}}{2})}. \tag{6}$$

In order to establish the correspondences across the target domain data y_t and the source domain data y_s, the maximum element in each column of $\mathbb{A}_c$ is preserved and set to 1 while the remaining elements are set to 0:

$$\mathbb{A}_c(i,j) = \begin{cases} 1, & if \quad \mathbb{A}_c(i,j) = max(\mathbb{A}_c(:,j)) \\ 0, & otherwise. \end{cases} \tag{7}$$

Given the set of transition matrixes for all the C categories, the global transition matrix $\mathbb{A}_c$ can be obtained by filling all the category-specific sub-matrixes into $\mathbb{A}_c$:

$$\mathbb{A} = \begin{pmatrix} \mathbb{A}_1 & & & \\ & \mathbb{A}_2 & & \\ & & \ddots & \\ & & & \mathbb{A}_C \end{pmatrix}, \tag{8}$$

where all the blank elements are set to 0, so that $\mathbb{A}$ is a reversible binary matrix. Assuming $\mathbb{A}$ leads to a perfect mapping across the sparse codes X_t and X_s and each matched pair of samples in different domains possesses an identical representation after encoding, then $\|X_t^T - \mathbb{A}X_s^T\|_F^2 = 0$, and correspondingly $\|Y_t^T - \mathbb{A}Y_s^T\|_F^2 = 0$. Thus Equation (3) can be rewritten as:

$$\begin{aligned} \langle D_t, D_s, X_t, X_s \rangle = arg \min_{D_t, D_s, X_t} \|Y_t - D_t X_t\|_2^2 \\ + \|(\mathbb{A}Y_s^T)^T - D_s X_t\|_2^2 \quad s.t. \forall i,\ \|x_t^i\|_0 \leq T. \end{aligned} \tag{9}$$

We attempt to further include a discriminative term to the objective function with respect to the optimal data distribution. Let the classifier $\mathcal{F}(x)$ satisfy the following equation:

$$\mathcal{P} = arg \min_{\mathcal{P}} \sum_i w_i \times \mathcal{L}\{h_i, \mathcal{F}(x_t^i, \mathcal{P})\} + \lambda_i \|\mathcal{P}\|_F^2, \tag{10}$$

where $\mathcal{L}$ is the classification loss function, h_i indicates the target domain labels of x_t^i, $\mathcal{P}$ denotes the classifier parameters and λ_i is a regularization parameter. As in previous work [17], [26], [10], [28], the classification error of a linear predictive classifier is included in the objective function:

$$\begin{aligned} &\langle D_t, D_s, X_t, \Phi, \mathcal{P} \rangle \\ &\quad = arg \min_{D_t, D_s, X_t, \Phi, \mathcal{P}} \|Y_t - D_t X_t\|_2^2 \\ &\quad + \alpha \|Q - \Phi X_t\|_2^2 + \beta \|\mathcal{H} - \mathcal{P} X_t\|_2^2 \\ &+ \|Y_s \mathbb{A}^T - D_s X_t\|_2^2 \quad s.t. \forall i,\ \|x_t^i\|_0 \leq T, \end{aligned} \tag{11}$$

where scalers α and β are set to control the relative contribution of the terms $\|Q - \Phi X_t\|_2^2$ and $\|\mathcal{H} - \mathcal{P} X_t\|_2^2$. Φ is a linear transformation matrix that maps the the original sparse codes to be in correspondence with the target discriminative sparse codes $Q = [q_1, q_2, \cdots, q_N] \in \Re^{K \times N}$ of the input signal Y_t. Specifically, $q_i = [q_i^1, q_i^2, \cdots, q_i^K] = [0, \cdots, 1, 1, \cdots, 0] \in \Re^K$, and the non-zeros occur at those indices where $y_t^i \in Y_t$ and $X_t^k \in X_t$ share the same class label. Given $X_t = [x_1, x_2, \cdots, x_6]$ and $Y_t = [y_1, y_2, \cdots, y_6]$, and assuming x_1, x_2, y_1 and y_2 are from class 1, x_3, x_4, y_3 and y_4 are from class 2, x_5, x_6, y_5 and y_6 are from class 3, Q is then defined with the following form:

$$\begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}, \tag{12}$$

and $\mathcal{H} = [h_1, h_2, \cdots, h_N] \in \Re^{C \times N}$ are the class labels of Y_t, where the non-zero element indicates the class of an input signal within each column $h_i = [0, \cdots, 1, \cdots, 0]^T \in \Re^C$. Following the same example in (12), $\mathcal{H}$ can be defined as:

$$\begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}. \tag{13}$$

2.3 Optimization

Let D_t and D_s have the same number of dictionary items, i.e., $K_t = K_s$, we rewrite Equation (11) as:

$$\begin{aligned} < D_t, D_s, X_t, \Phi, \mathcal{P} > = arg \min_{D_t, D_s, X_t, \Phi, \mathcal{P}} \\ \| \begin{pmatrix} Y_t \\ Y_s \mathbb{A}^T \\ \sqrt{\alpha} Q \\ \sqrt{\beta} H \end{pmatrix} - \begin{pmatrix} D_t \\ D_s \\ \sqrt{\alpha}\Phi \\ \sqrt{\beta}\mathcal{P} \end{pmatrix} X_t \|_2^2, \qquad s.t. \forall i, \ \|x_t^i\|_0 \leq T, \end{aligned} \tag{14}$$

We further define the left side of Equation (14) as $Y = (Y_t^T, (Y_s \mathbb{A}^T)^T, \sqrt{\alpha} Q^T, \sqrt{\beta} H^T)^T$ and the right side of Equation (14) as $D = (D_t^T, D_s^T, \sqrt{(\alpha)}\Phi^T, \sqrt{(\beta)}\mathcal{P}^T)^T$, where column-wise L_2 normalization is applied to D, so that Equation (14) can be efficiently solved by updating D and its corresponding coefficients X_t in an atom by column atom manner. Specifically, each column d_k and the corresponding x_t^k are optimized according to the following formulation:

$$< d_k, x_t^k > = arg \min_{d_k, x_t^k} \|E_k - d_k x_t^k\|_F^2, \quad s.t. \forall i, \ \|x_t^i\|_0 \leq T, \tag{15}$$

where $E_k = Y - \sum_{i \neq k} d_i * x_t^i$. The K-SVD algorithm [1] is adopted to solve such a problem:

$$\begin{aligned} U \Sigma V &= SVD(E_k) \\ \tilde{d}_k &= U(:,1) \\ \tilde{x}_t^k &= \Sigma(1,1) V(1,:), \end{aligned} \tag{16}$$

where $U(:,1)$ indicates the first column of U and $V(1,:)$ indicates the first row of V.

2.4 Classification

Since D_t, D_s, Φ and $\mathcal{P}$ are jointly normalized in the optimization procedure, they cannot be directly applied to construct the classification framework. Also, since $\mathcal{P}$ is obtained with the un-normalized D, simply re-normalizing D is not applicable. According to the lemma in

[28], $\tilde{D}_t$, $\tilde{D}_s$, $\tilde{\Phi}$ and $\tilde{\mathcal{P}}$ can be computed as:

$$\begin{aligned}
\tilde{D}_t &= \{\frac{d_t^1}{\|d_t^1\|_2}, \frac{d_t^2}{\|d_t^2\|_2}, \cdots, \frac{d_t^K}{\|d_t^K\|_2}\} \\
\tilde{D}_s &= \{\frac{d_s^1}{\|d_s^1\|_2}, \frac{d_s^2}{\|d_s^2\|_2}, \cdots, \frac{d_s^K}{\|d_s^K\|_2}\} \\
\tilde{\Phi} &= \{\frac{\phi^1}{\|\phi^1\|_2}, \frac{\phi^2}{\|\phi^2\|_2}, \cdots, \frac{\phi^K}{\|\phi^K\|_2}\} \\
\tilde{\mathcal{P}} &= \{\frac{p^1}{\|p^1\|_2}, \frac{p^2}{\|p^2\|_2}, \cdots, \frac{p^K}{\|p^K\|_2}\}
\end{aligned} \tag{17}$$

Given a target domain query sample y_t^i, its sparse representation x_t^i can be computed through $\tilde{D}_t$. With the linear classifier $\mathcal{F}(x:\tilde{\mathcal{P}})$, the label l_j of y_t^i can be predicted as:

$$l_j = arg\max_j (l = \tilde{\mathcal{P}} x_t^i). \tag{18}$$

3 Experiments

To demonstrate the effectiveness of our approach, experiments are conducted using two data sources, where the UCF YouTube action dataset [14] is treated as the target domain and the HMDB51 dataset [11] is treated as the source domain. Specifically, 7 body movements, including ride bike, dive, golf, jump, kick ball, ride horse and shoot ball, are chosen from the HMDB51 dataset in correspondence with similar actions in the UCF YouTube dataset. We run our method on five different partitions of the UCF YouTube dataset, where we randomly choose all action categories performed by the number of 5/9/16/20/24 actors as the training actions while using the remaining actions as the testing actions for each partition. Dense trajectories [22] are extracted from raw action video sequences with 8 spatial scales spaced by a factor of $1/\sqrt{2}$, and feature points are sampled on a grid spaced by 5 pixels and tracked in each scale, separately. Each point at frame t is tracked to the next frame $t+1$ by median filtering in a dense optical flow field. To avoid the drifting problem, the length of a trajectory is limited to 15 frames. HOG-HOF [12] and MBH [4] are computed within a $32\times32\times15$ volume along the dense trajectories, where each volume is sub-divided into a spatio-temporal grid of size $2\times2\times3$ to impose more structural information in the representation. Considering both efficiency and the construction error, the LLC coding scheme [23] is applied to the low-level local dense trajectory features with 30 local bases, and the codebook size is set to be 4000 for all training-testing partitions. To limit the complexity, only 200 local dense trajectory features are randomly selected from each video sequence when constructing the codebook. The weight α on the label constraint term and the weight β on the classification error term are set as 4 and 2 respectively, and 50 iterations of SVD decomposition are executed during optimization. We compare the performance of LLC, K-SVD [1] and LC-KSVD [10] with the proposed DCDDL method. Results are reported on both scenarios where the source domain data are included or excluded in TABLE 1. Among the listed methods, the dictionary learning process of K-SVD is unsupervised, and the dictionary learning process of LC-KSVD and DCDDL is supervised. When the source domain data are used by LLC, K-SVD and LC-KSVD, they are simply treated as extra training data without knowledge transfer. As shown in Figure 2, the proposed DCDDL method consistently leads to the best

Table 1: Performance comparison between DCDDL and other methods on the UCF YouTube dataset.

Algorithm	LLC	LLC	K-SVD	K-SVD	LC-KSVD	LC-KSVD	DCDDL
Learning	*N/A*	*N/A*	*Unsupervised*	*Unsupervised*	*Supervised*	*Supervised*	*Supervised*
Source data	*No*	*Yes*	*No*	*Yes*	*No*	*Yes*	*Yes*
24 actors	86.67%	86.67%	82.22%	77.78%	86.67%	82.22%	**88.89%**
20 actors	75.42%	70.21%	68.75%	72.08%	75.42%	75.42%	**77.50%**
16 actors	70.88%	70.17%	63.96%	67.54%	72.08%	72.08%	**73.03%**
09 actors	61.41%	61.80%	55.70%	59.15%	65.25%	64.72%	**66.31%**
05 actors	54.10%	53.35%	50.05%	48.88%	56.55%	54.10%	**56.66%**

performance over other methods under all dataset partitions. Note that the performance of LLC, K-SVD and LC-KSVD is even decreased when source domain data are used, which further validates the importance of our cross-domain dictionary learning.

Table 2: Performance comparison of DCDDL with state-of-the-art methods under the leave-one-actor-out setting on the UCF YouTube dataset.

Methods	[ ]	[ ]	BoF [ ]	DCDDL
Results	71.2%	75.21%	80.02%	**82.52%**

It is worth to point out that our dense trajectory features are simple concatenations of HOG, HOF and MBH, while Multiple Kernel Learning was adopted in [ ] to fuse these descriptors for the final SVM classification. Therefore, directly comparing with results in [ ] does not make any sense.

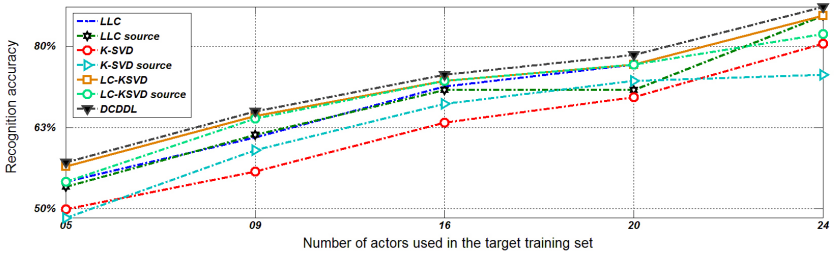

Figure 2: Performance comparison of the proposed DCDDL with other methods under different dataset partitions.

4 Conclusion

In this paper we have presented a novel cross-domain action recognition framework. Actions under mismatched data distributions can be unified into the same feature space through the discriminative cross-domain dictionary learning method, so that auxiliary domain knowledge can be utilized to span the intra-class diversities and improve the overall performance of the original recognition system. Through a transformation matrix, dictionary learning is performed on both the source domain actions and the target domain actions while no additional correspondence annotations between the two domains are required. Promising results are achieved on a realistic dataset, to which knowledge from a relevant dataset is transferred. The proposed framework can be easily adapted to solve other transfer learning problems and it leads to an interesting topic for future investigation when large scale source and target domain data are available.

References

[1] M. Aharon, M. Elad, and A. Bruckstein. K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *Signal Processing, IEEE Transactions on*, 54(1):4311–4322, 2006.

[2] Y. Boureau, F. Bach, Y. LeCun, and J. Ponce. Learning mid-level features for recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*. 2010.

[3] X. Cao, Z. Wang, F. Yan, and X. Li. Transfer learning for pedestrian detection. *Neurocomputing*, 100:51–57, 2012.

[4] N. Dalal, B. Triggs, and C. Schmid. Human detection using oriented histograms of flow and appearance. In *European Conference on Computer Vision*. 2006.

[5] L. Duan, I. W. Tsang, and D. Xu. Domain transfer multiple kernel learning. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(3):465–479, 2012.

[6] L. Duan, D. Xu, I. W. Tsang, and J. Luo. Visual event recognition in videos by learning from web data. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34 (9):1667–1680, 2012.

[7] L. Fei-Fei. Knowledge transfer in learning to recognize visual objects classes. In *International Conference on Development and Learning*. 2006.

[8] X. Gao, X. Wang, X. Li, and D. Tao. Transfer latent variable model based on divergence analysis. *Pattern Recognition*, 44:2358–2366, 2011.

[9] N. Ikizler-Cinbis and S. Sclaroff. Object, scene and actions: Combining multiple features for human action recognition. In *European Conference on Computer Vision*. 2010.

[10] Z. Jiang, Z. Lin, and L.S. Davis. Learning a discriminative dictionary for sparse coding via label consistent k-svd. In *IEEE Conference on Computer Vision and Pattern Recognition*. 2011.

[11] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. Hmdb: a large video database for human motion recognition. In *IEEE International Conference on Computer Vision*. 2011.

[12] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *IEEE Conference on Computer Vision and Pattern Recognition*. 2008.

[13] H. Lee, A. Battle, R. Raina, and A. Y. Ng. Efficient sparse coding algorithms. In *Advances in Neural Information Processing Systems*. 2007.

[14] J. Liu, J. Luo, and M. Shah. Recognizing realistic actions from videos "in the wild". In *IEEE Conference on Computer Vision and Pattern Recognition*. 2009.

[15] J. Liu, M. Shah, B. Kuipers, and S. Savarese. Cross-view action recognition via view knowledge transfer. In *IEEE Conference on Computer Vision and Pattern Recognition*. 2011.

[16] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Discriminative learned dictionaries for local image analysis. In *IEEE Conference on Computer Vision and Pattern Recognition*. 2008.

[17] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Supervised dictionary learning. In *Advances in Neural Information Processing Systems*. 2009.

[18] J. Marial, M. Leordeanu, F. Bach, M. Hebert, and J. Ponce. Discriminative sparse image models for class-specific edge detection and image interpretation. In *European Conference on Computer Vision*. 2008.

[19] C. Orrite, M. Rodríguez, and M. Montañés. One-sequence learning of human actions. *Human Behavior Understanding*, 7065:40–51, 2011.

[20] Q. Qiu, V. M. Patel, P. Turaga, and R. Chellappa. Domain adaptive dictionary learning. In *European Conference on Computer Vision*. 2012.

[21] R. Raina, A. Battle, H. Lee, B. Packer, and A. Y. Ng. Self-taught learning: transfer learning from unlabeled data. In *International Conference on Machine Learning*. 2007.

[22] H. Wang, A. Klaser, C. Schmid, and C. L. Liu. Action recognition by dense trajectories. In *IEEE Conference on Computer Vision and Pattern Recognition*. 2011.

[23] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong. Locality-constrained linear coding for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition*. 2010.

[24] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(2):210–227, 2009.

[25] J. Yang, R. Yan, and A. G. Hauptmann. Cross-domain video concept detection using adaptive svms. In *International Conference on Multimedia*. 2007.

[26] J. Yang, K. Yu, and T. Huang. Supervised translation-invariant sparse coding. In *IEEE Conference on Computer Vision and Pattern Recognition*. 2010.

[27] L. Yang, R. Jin, R. Sukthankar, and F. Jurie. Unifying discriminative visual codebook generation with classifier training for object category recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*. 2008.

[28] Q. Zhang and B. Li. Discriminative k-svd for dictionary learning in face recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*. 2010.

[29] W. Zhang, A. Surve, X. Fern, and T. Dietterich. Learning non-redundant codebooks for classifying complex objects. In *International Conference on Machine Learning*. 2009.

[30] J. Zheng, Z. Jiang, J. Phillips, and R. Chellappa. Cross-view action recognition via a transferable dictionary pair. In *British Machine Vision Conference*. 2012.

[31] M. Zhou, H. Chen, J. Paisley, L. Ren, G. Sapiro, and L. Carin. Non-parametric bayesian dictionary learning for sparse image representations. In *Advances in Neural Information Processing Systems*. 2009.