Cross-View Action Recognition via a Transferable Dictionary Pair

Jingjing Zheng[1]
zjngjng@umiacs.umd.edu
Zhuolin Jiang[2]
zhuolin@umiacs.umd.edu
P. Jonathon Phillips[3]
jonathon.phillips@nist.gov
Rama Chellappa[1]
rama@umiacs.umd.edu

[1] Department of Electrical and Computer Engineering and the Center for Automation Research, UMIACS University of Maryland College Park, MD, USA

[2] UMIACS, University of Maryland College Park, MD, USA

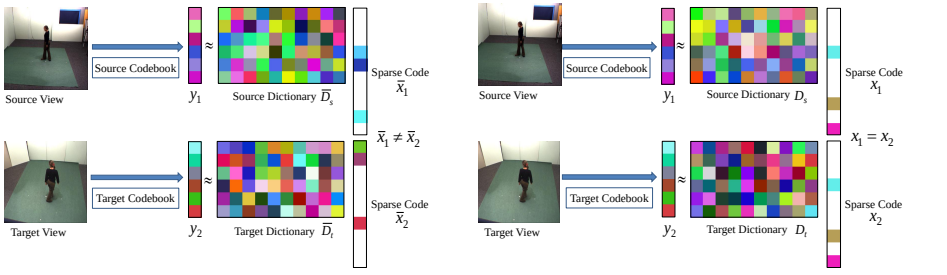
[3] National Institute of Standards and Technology Gaithersburg, MD, USA

Abstract

Discriminative appearance features are effective for recognizing actions in a fixed view, but generalize poorly to changes in viewpoint. We present a method for view-invariant action recognition based on sparse representations using a transferable dictionary pair. A transferable dictionary pair consists of two dictionaries that correspond to the source and target views respectively. The two dictionaries are learned simultaneously from pairs of videos taken at different views and aim to encourage each video in the pair to have the same sparse representation. Thus, the transferable dictionary pair links features between the two views that are useful for action recognition. Both unsupervised and supervised algorithms are presented for learning transferable dictionary pairs. Using the sparse representation as features, a classifier built in the source view can be directly transferred to the target view. We extend our approach to transferring an action model learned from multiple source views to one target view. We demonstrate the effectiveness of our approach on the multi-view IXMAS data set. Our results compare favorably to the the state of the art.

1 Introduction

Human action recognition is receiving significant attention in computer vision due to its rich real-world applications, which include multimedia retrieval, human computer interaction, video surveillance. Since many human actions produce strong spatio-temporal patterns of appearance or motion, most state-of-the-art approaches develop discriminative visual representations for recognizing actions. Some leading representations include spatio-temporal volumes [2, 30], spatio-temporal interest points [11, 15], shape features [3, 13, 16, 18] , geometric models of human body parts [19], and optical flow patterns [5, 13, 14]. These approaches are effective for recognizing actions with limited view variations but tend to perform poorly when applied on datasets with large view variations, such as actions in the



(a) Independent dictionary learning (b) Transferable dictionary learning

Figure 1: **Independent dictionary learning versus Transferable dictionary learning.** (a) Based on the BoVW feature representation, the source and target dictionaries are learned individually using videos taken from two different views of the same action. (b) Based on the same BoVW feature representation, we simultaneously learn the source and target dictionaries by forcing the shared videos taken from two views to have the same sparse representations.

IXMAS multi-view dataset [26] (See Fig. 2). This is because an action usually looks very different from different viewpoints. Thus, action models learned using labeled samples in one (source) view are less discriminative for recognizing actions in a different (target) view.

In this paper, we propose a novel approach for cross-view action recognition by transferring sparse feature representations of videos from the source to target view. The first step is to construct a separate codebook for each view, where the first view is the source domain and the second is the target domain. Each codebook is constructed by the k-means clustering algorithm. Each video is modeled as a Bag of Visual Words (BOVW) using the corresponding codebook from the same view. Although each pair of videos records the same action from two views, the feature representations of an action in the two views are different because each view has its own codebook. The next step is to learn a dictionary pair $\{D_s, D_t\}$, with D_s corresponding to the source view and D_t the target view. The dictionaries are designed to have sparse codes that are the same for each pair of videos that records the same action across the two views. In this way, videos across different views of the same action are encouraged to have similar sparse representations. This procedure enables the transfer of the sparse feature representations of videos in the source view to the corresponding videos in the target view. There is no reason to assume that two separate dictionaries that are learned independently for each view will have a view-invariant feature representation. The difference between learning a dictionary pair individually and our transferable dictionary pair learning can be seen in Figure 1.

Furthermore, we consider two types of actions: *shared* actions, that are observed in both *source* and *target* views, and *orphan* actions that are observed only in the source view. Orphan action labels are available only in the source view. For the shared actions, we consider two scenarios: (1) shared actions in both views are not labeled; (2) shared actions in both views are labeled. We refer them as the unsupervised and supervised settings respectively and propose corresponding unsupervised and supervised approaches for learning the transferable dictionary pair. Note that under both settings only videos of shared actions across different views are used for learning the dictionary pair, which means that the dictionary pair is not affected by videos of orphan actions. In order to handle the situation where videos of shared actions across multiple source views are available, we extend our approach to learn a set of view-dependent dictionaries where different BoVW model-based feature representations are converted to a dictionary set-based feature representations.

1.1 Our Contributions

We make the following contributions.

- Our approach directly exploits the video-level correspondence and bridges the gap of sparse representations of pairs of videos taken from different views of the same action.
- Our approach is unsupervised and does not require the category labels of the corresponding videos across two views. When the category labels are available, a discriminative dictionary pair can be learned to further improve the performance.
- We extend our approach to learn a set of dictionaries for transferring action models from multiple source views to one target view.
- Both unsupervised and supervised approaches are considered.

The paper is organized as follow. We briefly discuss related work in Section 2 and review sparse coding and dictionary learning methods in Section 3. Then we present our unsupervised and supervised approaches in Sections 4 and 5 respectively. Extensions to transfer action models from multiple source views to one target view are given in section 6. Experimental results are discussed in Section 7 and the paper is concluded in Section 8.

2 Related Work

Recently two transfer learning approaches have been proposed to address cross-view action recognition problems. Farhadi et al. [6] proposed a method that generates split-based features for frames in the source view using Maximum Margin Clustering and transfers the split values to the corresponding frames in the target view. From the split values of frames, a classifier is trained to predict split-based features in the target view. However, this approach requires feature-to-feature correspondence at the frame-level and the mapping from the original features to split-based features is obtained from a trained predictor. Liu et al. [17] used a bipartite graph to model the relationship between two codebooks generated by k-means clustering of videos acquired at each view. Then a bipartite partition is used to co-cluster the two view-dependent codebooks into shared visual-word clusters. A shared codebook made up of these shared clusters is used to encode all videos in both views. However, it only exploits the codebook-to-codebook correspondence at video-level, which can not guarantee that pairs of videos taken at the two views have similar feature representations based on the shard codebook. In addition, this method uses a fusion method to combine the prediction outputs of different transferred models. This requires the clustering of test videos in the target view.

Many other view-invariant methods that concentrate on the 2D image data acquired by multiple cameras have also been proposed. Rao et al [22] presented a view-invariant representation of human action to capture the dramatic changes in the speed and direction of the trajectory using spatio-temporal curvature of 2D trajectory. Parameswaran et al. [20, 21] proposed to model actions it terms of view-invariant canonical body poses and trajectories in 2D invariance space which leads to represent and recognize human actions from a general viewpoint. The approach in [9, 10] developed a very simple and stable action descriptor called Self-Similarity Matrix that captures the structure of temporal similarities and dissimilarities with an action sequence. The method in [7] proposed a latent model for cross-view action recognition which depends on good parameter initialization. The technique in [25] proposed a view-invariant matching method based on epipolar geometry between actor silhouettes without tracking and explicit point correspondences. Recently Li et al. [12] generated a sequence of linear transformations of action descriptors as smooth virtual path to connect the source view and target views.

In addition, 3D approaches for cross-view action recognition have also been proposed. One proposed method [26] models actions using three dimensional occupancy grids built from multiple view points, using an exemplar-based HMM. Yen et al. [29] employed a 4D action feature model for recognizing actions from arbitrary views. This model encodes shape and motion of actors observed from multiple views and requires the reconstruction of 3D visual hulls of actors at each time instant. Both approaches lead to computationally intense algorithms because finding the best match between a 3D model and a 2D observation requires searching over a large model parameter space. Weinland et al. [27] developed a hierarchical classification method based on 3D Histogram of Oriented Gradients (HOG) to represent a test sequence. Robustness to occlusions and viewpoint changes are achieved by combining training data from all viewpoints to train hierarchical classifiers.

3 Sparse Coding and Dictionary Learning

In this section, we give a brief review of sparse coding and the K-SVD algorithm [1] for learning an over-complete dictionary. Let $Y = [y_1, ..., y_N] \in \mathbb{R}^{n \times N}$ be a set of N input signals in a n-dimensional feature space. Assuming a dictionary D of size K is given, the sparse representations $X = [x_1, ..., x_N] \in \mathbb{R}^{K \times N}$ for Y are obtained by solving:

$$X = arg\min_X ||Y - DX||_2^2 \quad \text{s.t.} \quad \forall i, ||x_i||_0 \leq s, \tag{1}$$

where $||Y - DX||_2^2$ denotes the reconstruction error and $||x_i||_0 \leq s$ is the sparsity constraint. The sparsity constraint requires that each signal has s or fewer items in its decomposition. The orthogonal matching pursuit (OMP) algorithm [24] can then be used to solve Eq. 1.

The performance of sparse representation depends critically on D [8, 28]. The K-SVD [1] is well known for efficiently learning an over-complete dictionary from a set of training signals. It solves the following optimization problem:

$$(D, X) = arg\min_{D,X} ||Y - DX||_2^2 \quad \text{s.t.} \quad \forall i, ||x_i||_0 \leq s \tag{2}$$

where $D = [d_1, ..., d_K] \in \mathbb{R}^{n \times K}$ is the learned dictionary, and X are the sparse representations of Y. Later, we will formulate the problem of learning a transferable dictionary pair as an optimization problem which can be efficiently solved using the K-SVD algorithm.

4 Unsupervised Transferable Dictionary Pair Learning

In the unsupervised setting, our goal is to transfer orphan action models from the source view to the target view. In other words, we want to learn an action model for orphan actions in the source view and test it in the target view. We achieve this goal by making use of correspondence between two sets of videos of the shared unlabeled actions taken from two different views. Our solution is to find discriminative representations that are approximately the same for different views of the same action. For this purpose, we construct a transferable dictionary pair denoted by $\{D_s, D_t\}$, such that each pair of videos of the same action taken from source and target views have the same sparse representations. The key to the success of our method is that actions have sparse view-invariant representations, while each view has a different codebook.

Let $Y_s, Y_t \in \mathbb{R}^{n\times M}$ denote the feature representations of M videos of shared actions in the source and target views. The objective function for learning a transferrable dictionary pair is given by:

$$arg \min_{D_s,D_t,X} ||Y_s - D_s X||_2^2 + ||Y_t - D_t X||_2^2 \quad \text{s.t.} \quad \forall i, ||x_i||_0 \leq s. \tag{3}$$

Since we have the same number of shared action videos in source and target views, this objective function can be rewritten as

$$arg \min_{D,X} ||Y - DX||_2^2 \quad \text{s.t.} \quad \forall i, ||x_i||_0 \leq s \tag{4}$$

where $Y = \begin{bmatrix} Y_s \\ Y_t \end{bmatrix}$ and $D = \begin{bmatrix} D_s \\ D_t \end{bmatrix}$. As mentioned in Section 3, the transferable dictionary pair $\{D_s, D_t\}$ can be efficiently learned using the K-SVD algorithm.

Given the learned source dictionary D_s, we obtain sparse feature representations of the training videos in the source view using the OMP algorithm mentioned in Section 3. Similarly, for test videos of orphan actions in the target view, we obtain the corresponding sparse feature representations using D_t. Videos of orphan actions in both views will have similar sparse representations when encoded using the corresponding view-dependent dictionary. This is because D_s and D_t are learned by forcing two sets of videos of shared unlabeled actions in two views to have the same sparse representations. Thus, the action model learned in the source view can be directly applied to classify unlabeled test videos in the target view.

5 Supervised Transferable Dictionary Pair Learning

When action categories of shared action videos are available in both views, we leverage this category information to learn a discriminative transferrable dictionary pair. Here the key idea is to partition the total dictionary items into disjoint subsets and each subset is responsible for representing videos of one action. Specifically, we represent videos of the same action by the same subset of dictionary items. For videos of different action classes, we represent them using disjoint subsets of dictionary items. This results in an explicit correspondence between dictionary items and their labels. The rationale behind this idea is that action videos from the same class tend to have same features and each action video could be well represented by other videos from the same class. On the contrary, videos from different classes tend to have different features and thus should be well represented by disjoint subsets of other videos.

In order to achieve the above goal, we incorporate a label consistent regularization term introduced in [8] to the objective function in Eq. 3. Now the objective function for dictionary pair construction is given by:

$$arg \min_{D_s,D_t,A,X} ||Y_s - D_s X||_2^2 + ||Y_t - D_t X||_2^2 + \lambda ||Q - AX||_2^2 \quad \text{s.t.} \quad \forall i, ||x_i||_0 \leq s, \tag{5}$$

where λ controls the tradeoff between the reconstruction error and label consistent regularization. The elements of matrix $Q = [q_1, ..., q_N] \in \mathbb{R}^{K\times N}$ consist of the ideal "discriminative" sparse codes of shared action videos in both views. The vector $q_i = [q_i^1, ..., q_i^K] = [0...1, 1, ..0] \in \mathbb{R}^K$ is a discriminative sparse code corresponding to one shared action video pair $\{y_{s,i}, y_{t,i}\}$ and the non-zeros values of q_i occur at those indices where the shared action video pair $\{y_{s,i}, y_{t,i}\}$ and the dictionary item d_k share the same label. The matrix A is

a linear transformation matrix which transforms the original sparse code X to the be most discriminative in sparse feature space $\mathbb{R}^K$.

Similarly, the objective function in Eq. 5 can be rewritten as follows.

$$arg\min_{D,X} ||Y - DX||_2^2 \quad \text{s.t.} \quad \forall i, ||x_i||_0 \leq s \tag{6}$$

where $Y = \begin{bmatrix} Y_s \\ Y_t \\ Q \end{bmatrix}$ and $D = \begin{bmatrix} D_s \\ D_t \\ A \end{bmatrix}$. The K-SVD algorithm can be used to learn the transferable dictionary pair $\{D_s, D_t\}$. The learned transferable dictionary pair not only bridges the gap between sparse representations of action videos of the same class across two views, but also makes the sparse representations of action videos from different classes more discriminative.

6 Multi-view action recognition via multiple transferable dictionaries

In this section, we show how to extend our approach to transfer the action model from multiple source views to one target view. Suppose there are p source views $\mathcal{V}^s$ and one target view $\mathcal{V}^t$, the problem is how to make use of the transferred knowledge from each source view to recognize novel actions in the target view. We propose to learn a set of view-dependent dictionaries by forcing videos of shared actions in all views to have the same representations when encoded using the corresponding view-dependent dictionary. The corresponding objective function is given by:

$$arg \min_{\{D_{s,i}\}_{i=1}^p, D_t, X} \sum_{i=1}^{p} ||Y_{s,i} - D_{s,i}X||_2^2 + ||Y_t - D_tX||_2^2 \quad \text{s.t.} \quad \forall i, ||x_i||_0 \leq s. \tag{7}$$

Similarly, we rewrite the objective function as follows.

$$arg\min_{D,X} ||Y - DX||_2^2 \quad \text{s.t.} \quad \forall i, ||x_i||_0 \leq s, \tag{8}$$

where $Y = \begin{bmatrix} Y_{s,1} \\ \cdots \\ Y_{s,p} \\ Y_t \end{bmatrix}$ and $D = \begin{bmatrix} D_{s,1} \\ \cdots \\ D_{s,p} \\ D_t \end{bmatrix}$. It can be seen that our formulation not only aligns the correspondence between each source view and the target view but also aligns the correspondence among source views. Then we obtain the sparse representation of each video in each view using the corresponding view-dependent dictionary. Consequently, all videos in all views are aligned into a common view-invariant sparse feature space. This means that we do not need to differentiate the training videos from each source view in this common view-invariant sparse feature space. Any action model learned using all the training videos in all source views can be directly used to classify unlabeled test videos in the target view.

7 Experiments

We evaluated our approach using the IXMAS multi-view action data set [26], which contains four side views and one top view of 11 actions performed 3 times by 10 actors. See Figure 2 for example frames from the IXMAS dataset.

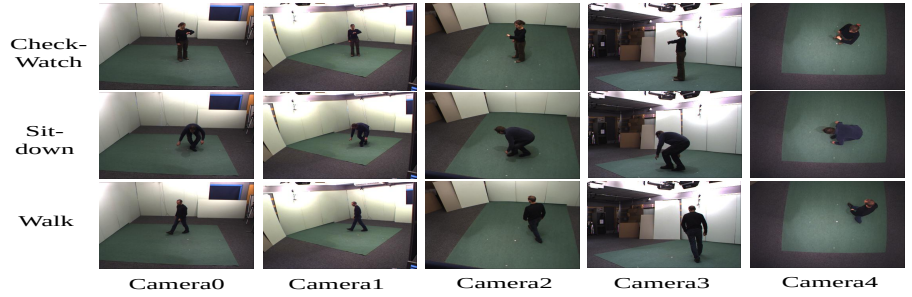


Figure 2: **Exemplar frames from IXMAS multi-view dataset.** Each row shows one action viewed across different angles.

We follow the protocol in [17] for extracting the spatio-temporal interest point feature introduced in [4]. We extract up to 200 cuboids from each action video. Each cuboid is represented by a 100-dimensional descriptor learned using PCA. Then we use these interest point descriptors to learn five codebooks of 1000 codewords by k-means clustering. One codebook is learned for each of the five views. Each action video is modeled as a BoVW using the corresponding view-dependent codebook. Thus, each action video is represented by a 1000-dimensional histogram. For global features, we learn five codebooks of 500 codewords by clustering shape flow descriptors introduced in [23] for each view. Similarly, each action video in each view is encoded using the corresponding view-dependent codebook. Finally, for each action video, we concatenate local and global feature descriptors to form a 1500-dimensional descriptor.

For an accurate comparison to [6] and [17], we follow the leave-one-action-class-out strategy for choosing the orphan action which means that each time we only consider one action class for testing in the target view. This action class is not used to construct a transferable dictionary pair in both unsupervised and supervised settings. We report the classification accuracy by averaging over all possible combinations for selecting orphan actions.

7.1 Transfer models across pairwise views

In this section, we evaluate our approach for transferring action models across pairwise views. We learn two dictionary pairs which consists of one independent dictionary pair and one transferable dictionary pair. Both dictionary pairs include two dictionaries corresponding to the source and target views. For the independent dictionary pair, we learn the dictionaries separately for each view without any knowledge transfer while the transferable dictionary pair is learned using Eq. 3 or Eq. 5 according to different settings. For each dictionary pair, we represent training videos in the source view and test videos in the target view using the corresponding source and target dictionaries respectively. Thus, we obtain two different sparse feature representations by using different dictionary pairs. Based on each sparse feature representation, a k-NN classifier is used to recognize unlabeled test videos. Table 1 shows the recognition accuracy for all 20 combinations of the source and target views. We observe that the k-NN without transfer performs very poorly and the recognition accuracy for most combinations is less than 50%. On the other hand, both of our approaches achieve very high accuracy, which demonstrates the transferability of the simultaneously learned dictionary pair.

The recognition results of different unsupervised and supervised approaches are shown in Table 2 and 3 respectively. Compared to the other two unsupervised approaches in [6]

%		Target View				
		C0	C1	C2	C3	C4
Source View	C0		(26.4, 96.7, 98.8)	(24.6, 97.9, 99.1)	(20.3, 97.6, 99.4)	(27.9, 84.9, 92.7)
	C1	(31.2, 97.3, 98.8)		(23.0, 96.4, 99.7)	(23.0, 89.7, 92.7)	(20.3, 81.2, 90.6)
	C2	(23.3, 92.1, 99.4)	(20.9, 89.7, 96.4)		(13.0, 94.9, 97.3)	(17.9, 89.1, 95.5)
	C3	(9.7, 97.0, 98.2)	(24.9, 94.2, 97.6)	(23.0, 96.7, 99.7)		(16.7, 83.9, 90.9)
	C4	(51.2, 83.0, 85.8)	(38.2, 70.6, 81.5)	(41.2, 89.7, 93.3)	(53.3, 83.7, 83.9)	
	Ave.	(28.9, 92.4, 95.5)	(27.6, 87.8, 93.6)	(28.0, 95.1, 98.0)	(27.4, 91.2, 93.3)	(20.7, 84.8, 92.4)

Table 1: **The cross-view recognition results with and without knowledge transfer.** Each row corresponds to a source (training) view and each column a target (test) view. $\{Ci\}_{i=0,1,...,4}$ denotes five different camera views. The recognition numbers in the bracket are the average recognition accuracies of k-NN without transfer, our unsupervised and supervised approaches respectively.

%		Target View				
		C0	C1	C2	C3	C4
Source View	C0		(72, 77.6, 79.9, 96.7)	(61, 69.4, 76.8, 97.9)	(62, 70.3, 76.8, 97.6)	(30, 44.8, 74.8, 84.9)
	C1	(69, 77.3, 81.2, 97.3)		(64, 73.9, 75.8, 96.4)	(68, 67.3, 78.0, 89.7)	(41, 43.9, 70.4, 81.2)
	C2	(62, 66.1, 79.6, 92.1)	(67, 70.6, 76.6, 89.7)		(67, 63.6, 79.8, 94.9)	(43, 53.6, 72.8, 89.1)
	C3	(63, 69.4, 73.0, 97.0)	(72, 70.0, 74.1, 94.2)	(51, 51.8, 74.0, 96.7)		(44, 44.2, 66.9, 83.9)
	C4	(51, 39.1, 82.0, 83.0)	(55, 38.8, 68.3, 70.6)	(51, 51.8, 74.0, 89.7)	(53, 34.2, 71.1, 83.7)	
	Ave.	(61, 63.0, 79.0, 92.4)	(67, 64.3, 74.7, 87.8)	(61, 64.5, 75.2, 95.1)	(63, 58.9, 76.4, 91.2)	(40, 46.6, 71.2, 84.8)

Table 2: **The cross-view recognition results of different unsupervised approaches.** Each row corresponds to a source (training) view and each column a target (test) view. $\{Ci\}_{i=0,1,...,4}$ denotes five different camera views. The recognition numbers in the bracket are the average recognition accuracies of [6], [10], [17], and our unsupervised approach respectively.

%		Target View				
		C0	C1	C2	C3	C4
Source View	C0		(79, 98.8)	(79, 99.1)	(68, 99.4)	(76, 92.7)
	C1	(72, 98.8)		(74, 99.7)	(70, 92.7)	(66, 90.6)
	C2	(71, 99.4)	(82, 96.4)		(76, 97.3)	(72, 95.5)
	C3	(75, 98.2)	(75, 97.6)	(73, 99.7)		(76, 90.0)
	C4	(80, 85.8)	(73, 81.5)	(73, 93.3)	(79, 83.9)	
	Ave.	(74, 95.5)	(77, 93.6)	(76, 98.0)	(73, 93.3)	(72, 92.4)

Table 3: **The cross-view recognition results of different supervised approaches.** Each row corresponds to a source (training) view and each column a target (test) view. $\{Ci\}_{i=0,1,...,4}$ denotes five different camera views. The recognition numbers in the bracket are the average recognition accuracies of [7] and our supervised approach respectively.

and [17] that also transfer action models across views, our unsupervised approach yields much better performance in all cases. Furthermore, for half of all the combinations, our unsupervised method achieves more than 90% recognition accuracy. Comparing our supervised approach with [7], which also requires supervision, our method still performs much better and achieves nearly perfect performance for a majority of combinations.

It is interesting to note that for the case where the Camera4 is the source or the target view, the recognition accuracy is a little lower than other combinations of piecewise views. One reason for this is that Camera 4 was set above the actors and all the different actions look the same from the top view. In addition, the higher recognition accuracy obtained by our supervised approach compared to the unsupervised approach demonstrates that the transferable dictionary pair learned using labeled shared activities across views is more discriminative.

7.2 Transfer models from multiple source views to one target view

In this section, we show the effectiveness of our approach in transferring action models from multiple source views to one target view. We learn a dictionary set $\{D_{s,1},...,D_{s,p},D_t\}$ according to Eq. 7 in the unsupervised learning. For the supervised setting, we add the label consistent regularization term to Eq. 7 to learn a more discriminative dictionary set. Afterwards, each training action video in each view is encoded using the corresponding view-dependent dictionary respectively. Since the dictionary set is learned by aligning the

Percentage	Camera0	Camera1	Camera2	Camera3	Camera4	Average
Our unsupervised	**98.5**	**99.1**	**99.1**	**100**	**90.3**	**97.4**
Our supervised	**99.4**	**98.8**	**99.4**	**99.7**	**93.6**	**98.2**
LWE [12]	86.6	81.1	80.1	83.6	82.8	82.8
Junejo et al. [10]	74.8	74.5	74.8	70.6	61.2	71.2
Liu et al. [15]	76.7	73.3	72.0	73.0	N/A	73.8
Weinland et al. [27]	86.7	89.9	86.4	87.6	66.4	83.4

Table 4: **Multi-view action recognition results .** Each column corresponds to one target view. The first two rows show the recognition accuracies of our unsupervised and supervised approaches.

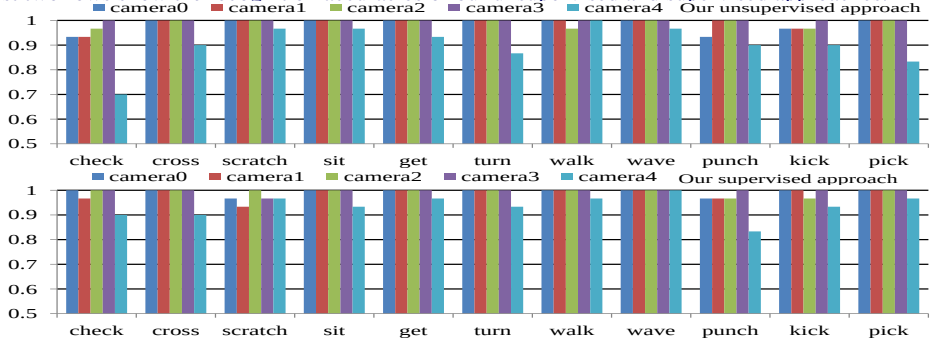


Figure 3: **The multi-view recognition results on each action category.** The top bar figure corresponds to our unsupervised approach and the bottom bar figure corresponds to our supervised approach. Five bars in a group indicates the recognition accuracy for one action category and each bar in the group corresponds to one target (test) view.

correspondence of shared action videos across all views, the sparse representation of all action videos in all views are in the same sparse feature space. And we simply use a k-NN classifier to classify the unlabeled test videos in the target view.

Table 4 shows the average accuracy for transferring action models from multiple source views to one target view. It can be seen that both unsupervised and supervised approaches presented here obtain the best performance and achieve nearly perfect performance for all cases except the case where Camera 4 (top view) is the target view. Furthermore, both [12] and our unsupervised approach only use training videos of four source views to train a classifier while [10, 15, 27] trained their classifiers using training videos from all five views.

In addition, it is interesting to look at the recognition accuracy of each action category from each target view in Figure 3. We observe that except for the Camera 4 (top view) as the target view, both of our approaches achieves more than 90% accuracy for each action category in each target view. This again shows that it is harder to transfer action models across views that involves the top view.

8 Conclusion

In this paper, we introduced a dictionary learning-based approach to recognize an unknown action from an unseen (target) view using training data taken from other (source) views. We propose to learn a transferable dictionary pair which includes a source dictionary and a target dictionary using shared action videos across the source and target views. By forcing the shared action videos in both views to have the same sparse representations, the dictionary pair is made to have the transferability property. This is because action videos of the same class in different views encoded using the corresponding view-dependent dictionary tend to have the same sparse representations. Using this transferable dictionary pair, we can directly transfer action models across views. In addition, when the labels of shared action videos are

available in both views, we extend our approach to learn a more discriminative transferable dictionary pair by forcing shared action videos of different classes inside each view to have different sparse representations. Furthermore, we naturally extend our approach to transfer action models from multiple source views to one target view without using model fusion methods. We have extensively tested our approach on the publicly available IXMAS multi-view dataset. The resulting performance clearly confirms the effectiveness of our approach for cross-view action recognition.

9 Acknowledgement

The first, second and fourth authors were supported by a MURI grant N00014-10-1-0934 from the Office of Naval Research.

References

[1] Michal Aharon, Michael Elad, and Alfred Bruckstein. K -SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *Signal Processing, IEEE Transactions on*, 2006.

[2] Moshe Blank, Lena Gorelick, Eli Shechtman, Michal Irani, and Ronen Basri. Actions as space-time shapes. In *ICCV*, 2005.

[3] German K. M. Cheung, Simon Baker, and Takeo Kanade. Shape-from-silhouette of articulated objects and its use for human body kinematics estimation and motion capture. In *CVPR*, 2003.

[4] Piotr Dollár, Vincent Rabaud, Garrison Cottrell, and Serge Belongie. Behavior recognition via sparse spatio-temporal features. In *VS-PETS*.

[5] Alexei A. Efros, Alexander C. Berg, Greg Mori, and Jitendra Malik. Recognizing action at a distance. In *ICCV*, 2003.

[6] Ali Farhadi and Mostafa Kamali Tabrizi. Learning to recognize activities from the wrong view point. In *ECCV*, 2008.

[7] Ali Farhadi, Mostafa Kamali Tabrizi, Ian Endres, and David A. Forsyth. A latent model of discriminative aspect. In *ICCV*, 2009.

[8] Zhuolin Jiang, Zhe Lin, and Larry S. Davis. Learning a discriminative dictionary for sparse coding via label consistent K-SVD. In *CVPR*, 2011.

[9] Imran N. Junejo, Emilie Dexter, Ivan Laptev, and Patrick Pérez. View-independent action recognition from temporal self-similarities. *IEEE Trans. Pattern Anal. Mach. Intell.*

[10] Imran N. Junejo, Emilie Dexter, Ivan Laptev, and Patrick Pérez. Cross-view action recognition from temporal self-similarities. In *ECCV*, 2008.

[11] Ivan Laptev and Tony Lindeberg. Space-time interest points. In *ICCV*, 2003.

[12] Ruonan Li and Todd Zickler. Discriminative virtual views for cross-view action recognition. In *CVPR*, 2012.

[13] Zhe Lin, Zhuolin Jiang, and Larry S. Davis. Recognizing actions by shape-motion prototype trees. In *ICCV*, 2009.

[14] Jim Little and Jeffrey E. Boyd. Recognizing people by their gait: The shape of motion. *Videre*, 1996.

[15] Jingen Liu and Mubarak Shah. Learning human actions via information maximization. In *CVPR*, 2008.

[16] Jingen Liu, Saad Ali, and Mubarak Shah. Recognizing human actions using multiple features. In *CVPR*, 2008.

[17] Jingen Liu, Mubarak Shah, Benjamin Kuipers, and Silvio Savarese. Cross-view action recognition via view knowledge transfer. In *CVPR*, 2011.

[18] Fengjun Lv and Ramakant Nevatia. Single view human action recognition using key pose matching and viterbi path searching. In *CVPR*, 2007.

[19] Greg Mori, Xiaofeng Ren, Alexei A. Efros, and Jitendra Malik. Recovering human body configurations: Combining segmentation and recognition. In *CVPR*, 2004.

[20] Vasu Parameswaran and Rama Chellappa. Human action-recognition using mutual invariants. *Computer Vision and Image Understanding*, 2005.

[21] Vasu Parameswaran and Rama Chellappa. View invariance for human action recognition. *International Journal of Computer Vision*, 2006.

[22] Cen Rao, Alper Yilmaz, and Mubarak Shah. View-invariant representation and recognition of actions. *International Journal of Computer Vision*, 2002.

[23] Du Tran and Alexander Sorokin. Human activity recognition with metric learning. In *ECCV*, 2008.

[24] Joel A. Tropp and Anna C. Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Transactions on Information Theory*.

[25] Anwaar ul Haq, Iqbal Gondal, and Manzur Murshed. On dynamic scene geometry for view-invariant action matching. In *CVPR*, 2011.

[26] Daniel Weinland, Edmond Boyer, and Rémi Ronfard. Action recognition from arbitrary views using 3D exemplars. In *ICCV*, 2007.

[27] Daniel Weinland, Mustafa Özuysal, and Pascal Fua. Making action recognition robust to occlusions and viewpoint changes. In *ECCV*, 2010.

[28] John Wright, Yi Ma, Julien Mairal, Guillermo Sapiro, Thomas S. Huang, and Shuicheng Yan. Sparse representation for computer vision and pattern recognition. *Proceedings of the IEEE*, 2010.

[29] Pingkun Yan, Saad M. Khan, and Mubarak Shah. Learning 4d action feature models for arbitrary view action recognition. In *CVPR*, 2008.

[30] Alper Yilmaz and Mubarak Shah. Actions sketch: A novel action representation. In *CVPR*, 2005.