Learning View-invariant Sparse Representations for Cross-view Action Recognition

Jingjing Zheng[†], Zhuolin Jiang[§]
[†]University of Maryland, College Park, MD, USA
[§]Noah's Ark Lab, Huawei Technologies
zjngjng@umiacs.umd.edu, zhuolin.jiang@huawei.com

Abstract

We present an approach to jointly learn a set of view-specific dictionaries and a common dictionary for cross-view action recognition. The set of view-specific dictionaries is learned for specific views while the common dictionary is shared across different views. Our approach represents videos in each view using both the corresponding view-specific dictionary and the common dictionary. More importantly, it encourages the set of videos taken from different views of the same action to have similar sparse representations. In this way, we can align view-specific features in the sparse feature spaces spanned by the view-specific dictionary set and transfer the view-shared features in the sparse feature space spanned by the common dictionary. Meanwhile, the incoherence between the common dictionary and the view-specific dictionary set enables us to exploit the discrimination information encoded in view-specific features and view-shared features separately. In addition, the learned common dictionary not only has the capability to represent actions from unseen views, but also makes our approach effective in a semi-supervised setting where no correspondence videos exist and only a few labels exist in the target view. Extensive experiments using the multi-view IXMAS dataset demonstrate that our approach outperforms many recent approaches for cross-view action recognition.

1. Introduction

Action recognition has many potential applications in multimedia retrieval, video surveillance and human computer interaction. In order to accurately recognize human actions, most existing approaches focus on developing different discriminative features, such as spatio-temporal interest point (STIP) based features [32, 2, 13, 19], shape [16, 3, 21] and optical flow based features [5, 17, 16]. These features are effective for recognizing actions taken from similar viewpoints, but perform poorly when viewpoints vary significantly. Extensive experiments in [20, 33] have shown that failing to handle feature variations caused by viewpoints may yield inferior results. This is because the same action looks quite different from different viewpoints as shown in Figure 1. Thus action models learned from one view become less discriminative for recognizing actions in a much different view.

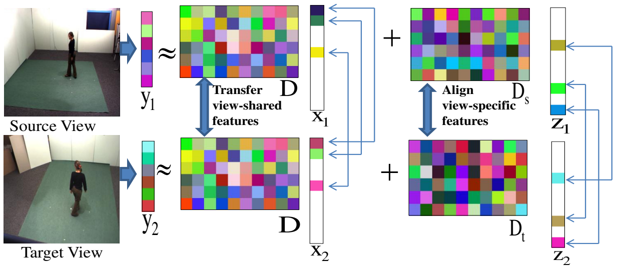


Figure 1. **Joint learning of a view-specific dictionary pair and a common dictionary.** We not only learn a common dictionary D to model view-shared features of corresponding videos in both views, but also learn two view-specific dictionaries D^s and D^t that are incoherent to D to align the view-specific features. The sparse representations ($x1$ and $x2$, $z1$ and $z2$) share the same sparsity patterns (selecting the same items).

A very fruitful line of work for cross-view action recognition based on transfer learning is to construct the mappings or connections between different views, by using videos taken from different views of the same action [6, 7, 20, 8]. [6] exploited the frame-to-frame correspondence in pairs of videos taken from two views of the same action by transferring the split-based features of video frames in the source view to the corresponding video frames in the target view. [20] proposed to exploit the correspondence between the view-dependent codebooks constructed by k-means clustering on videos in each view. However, the frame-to-frame correspondence [6] is computationally expensive, and the codebook-to-codebook correspondence [20] is not accurate enough to guarantee that a pair of videos observed in the source and target views will have similar feature representations.

In order to overcome these drawbacks, we propose a dictionary learning framework to exploit the video-to-video correspondence by encouraging pairs of videos taken in two

This research was supported by a MURI from the US Office of Naval Research under the Grant N00014-10-1-0934.

views to have similar sparse representations. Figure 1 illustrates our dictionary learning framework. Our approach not only learns a common dictionary shared by different views to model the view-shared features, but also learns a dictionary pair corresponding to the source and target views to model and align view-specific features in the two views. Both the common dictionary and the corresponding view-specific dictionary are used to represent videos in each view. Instead of transferring the split-features as in [6], we transfer the indices of the non-zero elements (i.e., the indices of selected dictionary items) in sparse codes of videos from the source view to sparse codes of the corresponding videos from the target view. In other words, we not only use the same subset of dictionary items from the common dictionary to represent view-shared features in correspondence videos from different views, but also use the same subset of dictionary items from different view-specific dictionaries to represent view-specific features. In this way, videos across different views of the same action tend to have similar sparse representations. Note that our approach enforces the common dictionary to be incoherent with view-specific dictionaries, so that the discrimination information encoded in view-specific features and view-shared features are exploited separately and makes view-specific dictionaries more compact.

Actions are categorized into two types: *shared* actions observed in both views and *orphan* actions that are only observed in the source view. Note that only pairs of videos taken from two views of the shared actions are used for dictionary learning. In addition, we consider two scenarios for the shared actions: (1) shared actions in both views are unlabeled. (2) shared actions in both views are labeled. These two scenarios are referred to as unsupervised and supervised settings, respectively, in subsequent discussions.

1.1. Contributions

The main contributions of this paper are:

- We propose to simultaneously learn a set of view-specific dictionaries to exploit the video-level correspondence across views and a common dictionary to model the common patterns shared by different views.
- The incoherence between the common dictionary and the view-specific dictionaries enables our approach to drive the shared pattern to the common dictionary and focus on exploiting the discriminative correspondence information encoded by the view-specific dictionaries.
- With the separation of the common dictionary, our approach not only learns more compact view-specific dictionaries, but also bridges the gap of the sparse representations of correspondence videos taken from different views of the same action using a more flexible method.
- Our framework is a general approach and can be applied to cross-view and multi-view action recognition under both unsupervised and supervised settings.

2. Related Work

Recently, several transfer learning techniques have been proposed for cross-view action recognition [6, 20, 8, 33]. Specifically, [6] proposed to generate the same split-based features for correspondence video frames from both the source and target views. It is computationally expensive because it requires the construction of feature-to-feature correspondence at the frame-level and learning an additional mapping from original features to the split-based features. [20] used a bipartite graph to model the relationship between two view-dependent codebooks. Even though this approach exploits the codebook-to-codebook correspondence between two views, it can not guarantee that videos taken at different views of shared actions will have similar features. [8] used canonical correlation analysis to derive a correlation subspace as a joint representation from different bag-of-words models at different views and incorporate a corresponding correlation regularizer into the formulation of support vector machine. [33] proposed a dictionary learning framework for cross-view action recognition with the assumption that sparse representations of videos from different views of the same action should be strictly equal. However, this assumption is too strong to flexibly model the relationship between different views.

Many view-invariant approaches that use 2D image data acquired by multiple cameras have also been proposed. [25, 22, 23] proposed view-invariant representations based on view-invariant canonical body poses and trajectories in 2D invariance space. [11, 10] captured the structure of temporal similarities and dissimilarities within an action sequence using a Self-Similarity Matrix. [27] proposed a view-invariant matching method based on epipolar geometry between actor silhouettes without tracking and explicit point correspondences. [15] learned two view-specific transformations for the source and target views, and then generated a sequence of linear transformations of action descriptors as the virtual views to connect two views. [14] proposed the Hankel matrix of a short tracklet which is a view-invariant feature to recognize actions across different viewpoints.

Another fruitful line of work for cross-view action recognition concentrates on using the 3D image data. The method introduced in [28] employed three dimensional occupancy grids built from multi-view points to model actions. [31] developed a 4D view-invariant action feature extraction to encode the shape and motion information of actors observed from multiple views. Both of these approaches lead to computationally intense algorithms because they need to find the best match between a 3D model and a 2D observation over a large model parameter space. [29] developed a robust and view-invariant hierarchical classification method based on 3D HOG to represent a test sequence.

3. Learning View-invariant Sparse Representations via Dictionary Learning

3.1. Unsupervised Learning

In the unsupervised setting , our goal is to find view-invariant feature representations by making use of correspondence between videos of the shared actions taken from different views. Let $Y^v = [y_1^v, ..., y_N^v] \in \mathbf{R}^{d\times N}$ denote d-dimensional feature representations of N videos of the shared actions taken in the v-th view. $Y_i = [y_i^1, ..., y_i^V]$ are V action videos of the shared action y_i taken from V views, which are referred to as *correspondence* videos. On one hand, we would like to learn a common dictionary $D \in \mathbf{R}^{d\times J}$ with a size of J shared by different views to represent videos from all views. On the other hand, for each view, we learn $D^v \in \mathbf{R}^{d\times J^v}$ to model the view-specific features. The objective function for the unsupervised setting is:

$$\begin{aligned}&\sum_{i=1}^{N}\{\sum_{v=1}^{V}\{||y_i^v - Dx_i^v||_2^2 + ||y_i^v - Dx_i^v - D^v z_i^v||_2^2\}\\&+ \lambda||X_i||_{2,1} + \lambda||Z_i||_{2,1}\} + \eta\sum_{v=1}^{V}||D^T D^v||_F^2\end{aligned} \tag{1}$$

where $X_i = [x_i^1, ..., x_i^V], Z_i = [z_i^1, ..., z_i^V]$ are the joint sparse representations for y_i across V views. This objective function consists of five terms:

1. The first two terms are the reconstruction errors of videos from different views using D only or using both D and D^v. The minimization of the first reconstruction error enables D to encode view-shared features as much as possible while the minimization of the second reconstruction error enables D^v to encode and align view-specific features that can not be modeled by D.
2. The third and fourth terms are the sparse representations via $L_{2,1}$-norm regularization using D and D^v respectively. The $L_{2,1}$-norm minimization for X and Z can make the entries in each row of the two matrices to be all zeros or non-zeros at the same time. This means that we not only encourage to use the same subset of dictionary items in D to represent the correspondence videos from different views, but also encourage to use dictionary items from D^v with the same index of selected dictionary items to further reduce the reconstruction error of videos in each view. Therefore the testing videos taken from different views of the same action will be encouraged to have similar sparse representations when using the learned D and D^v.
3. The last term regularizes the common dictionary to be incoherent to the view-specific dictionaries. The incoherence between D and D^v enables our approach to separately exploit the discriminative information encoded in the view-specific features and view-shared features.

3.2. Supervised Learning

Given the action categories of correspondence videos, we can learn a discriminative common dictionary and discriminative views-specific dictionaries by leveraging the category information. We partition the dictionary items in each dictionary into disjoint subsets and associate each subset with one specific class label. For videos from action class k, we aim to represent them using the same subset of dictionary items associated with class k. For videos from different classes, we represent them using disjoint subsets of dictionary items. This is supported by the intuition that action videos from the same class tend to have the similar features and each action video can be well represented by other videos from the same class [30]. We incorporate the discriminative sparse code error term introduced in [9] to achieve this goal.

Assume there are K shared action classes, and $D = [D_1, ..., D_K]$, $D^v = [D_1^v, ..., D_K^v]$ where $D_k \in \mathbf{R}^{d\times J_k}$,$\sum_{k=1}^{K} J_k = J$, and $D_k^v \in \mathbf{R}^{d\times J_k^v}$, $\sum_{k=1}^{K} J_k^v = J^v$, the objective function for the supervised setting is:

$$\begin{aligned}&\sum_{i=1}^{N}\{\sum_{v=1}^{V}\{||y_i^v - Dx_i^v||_2^2 + ||y_i^v - Dx_i^v - D^v z_i^v||_2^2\\&+ ||q_i - Ax_i^v||_2^2 + ||q_i^v - Bz_i^v||_2^2\} + \lambda||X_i||_{2,1}\\&+ \lambda||Z_i||_{2,1}\} + \eta\sum_{v=1}^{V}||D^T D^v||_F^2\end{aligned} \tag{2}$$

where $q_i = [q_{i_1}, ..., q_{i_K}]^T \in \mathbf{R}^{J\times 1}$ and $q_i^v = [q_{i_1}^v, ..., q_{i_K}^v]^T \in \mathbf{R}^{J^v\times 1}$ called 'discriminative' sparse coefficients associated with D and D^v respectively. When a video y_i^v is from class k at the v-th view, then q_{i_k} and $q_{i_k}^v$ are ones and other entries in q_i and q_i^v are zeros. $A \in \mathbf{R}^{J\times J}$ and $B \in \mathbf{R}^{J^v\times J^v}$ are called transformation matrices which transform x_i^v and z_i^v to approximate q_i and q_i^v respectively. The discriminative sparse-code error terms $||q_i - Ax_i^v||_2^2$ and $||q_i^v - Bz_i^v||_2^2$ encourage the dictionary items with class k to be selected to reconstruct those videos from class k. Note that the $L_{2,1}$-norm regularization only regularize the relationship between the sparse codes of correspondence videos, but can not regularize the relationship between the sparse codes of videos from the same action class in each view. The integration of discriminative sparse code error term in the objective function can address this issue. In other words, our approach not only encourages the videos taken from different views of the same action to have similar sparse representations, but also encourages videos from the same class in each view to have similar sparse representations.

3.3. Optimization

Here we only describe the optimization of the objective function in (2) while the optimization of (1) utilizes the similar procedure except that A and B components are excluded. This optimization problem is divided into three subproblems: (1) computing sparse codes with fixed D^v, D and A, B; (2) updating D^v, D with fixed sparse codes and A, B; (3) updating A, B with fixed D^v, D and sparse codes.

3.4. Computing Sparse Codes

Given fixed D^v, D and A, B, we solve the sparse coding problem of the correspondence videos set by set and (2) is reduced to:

$$\sum_{v=1}^{V}\{||y_i^v - Dx_i^v||_2^2 + ||y_i^v - Dx_i^v - D^v z_i^v||_2^2 + ||q_i - Ax_i^v||_2^2 \\ + ||q_i^v - Bz_i^v||_2^2\} + \lambda||X_i||_{2,1} + \lambda||Z_i||_{2,1}\}. \tag{3}$$

We rewrite (3) as follows:

$$\sum_{v=1}^{V}||\tilde{y}_i^v - \tilde{D}^v \tilde{z}_i^v||_2^2 + \lambda||\tilde{Z}_i||_{2,1} \tag{4}$$

where $\tilde{y}_i^v = \begin{bmatrix} y_i^v \\ y_i^v \\ q_i \\ q_i^v \end{bmatrix}, \tilde{D^v} = \begin{bmatrix} D & O_1 \\ D & D^v \\ A & O_2 \\ O_3 & B \end{bmatrix}, \tilde{z}_i^v = \begin{bmatrix} x_i^v \\ z_i^v \end{bmatrix}, \tilde{Z}_i = [\tilde{z}_i^1, ..., \tilde{z}_i^V]$ and $O_1 \in \mathbf{R}^{d\times J^v}, O_2 \in \mathbf{R}^{J\times J^v}, O_3 \in \mathbf{R}^{J^v\times J}$ are matrices of all zeros. The minimization of (4) is known as a multi-task group lasso problem [18] where each view is treated as a task. We use the software SLEP in [18] for computing sparse codes.

3.5. Updating Dictionaries

Given fixed sparse codes and A, B, (2) is reduced to:

$$\sum_{i=1}^{N}\sum_{v=1}^{V}\{||y_i^v - Dx_i^v||_2^2 + ||y_i^v - Dx_i^v - D^v z_i^v||_2^2\} \\ + \eta\sum_{v=1}^{V}||D^T D^v||_F^2 \tag{5}$$

We rewrite (5): $\sum_{v=1}^{V}\{||Y^v - DX^v||_F^2 + ||Y^v - DX^v - D^v Z^v||_F^2\} + \eta\sum_{v=1}^{V}||D^T D^v||_F^2$ where $Y^v = [y_1^v, ..., y_N^v], X^v = [x_1^v, ..., x_N^v], Z^v = [z_1^v, ..., z_N^v]$. Motivated by [12], we first fix D^v and then update $D = [d_1, ..., d_J]$ atom by atom, i.e. updating d_j while fixing other column atoms in D. Specifically, let $\hat{Y}^v = Y^v - \sum_{m\neq j} d_m x_{(m)}^v$ where $x_{(m)}^v$ corresponds to the m-th row of X^v, we solve the following problem for updating d_j in D: $arg\min_{d_j} f(d_j) = \sum_{v=1}^{V}\{||\hat{Y}^v - d_j x_{(j)}^v||_F^2 + ||\hat{Y}^v - D^v Z^v - d_j x_{(j)}^v||_F^2 + \eta||d_j^T D^v||_F^2$. Let the first-order derivative of $f(d_j)$ with respect to d_j equal to zero, *i.e.* $\frac{\partial f(d_j)}{\partial d_j} = 0$, then we can update d_j as:

$$d_j = \frac{1}{2}\sum_{v=1}^{V}(||x_{(j)}^v||_2^2 I + \frac{\eta}{2}D^v D^{vT})^{-1}(2\hat{Y}^v - D^v Z^v)x_{(j)}^{vT}. \tag{6}$$

Now we fix D and update D^v atom by atom. Each item d_j^v in D^v is updated as :

$$d_j^v = \frac{1}{2}(||z_{(j)}^v||_2^2 I + \frac{\eta}{2}DD^T)^{-1}\bar{Y}^v z_{(j)}^{vT}. \tag{7}$$

where $\bar{Y}^v = Y^v - DX^v - \sum_{m\neq j} d_m^v z_{(m)}^v$.

3.6. Updating A, B

Given sparse codes and all the dictionaries, we employ the multivariate ridge regression model [24] to update A, B with the quadratic loss and l_2 norm regularization:

$$\min_A \sum_{i=1}^{N}\sum_{v=1}^{V}||q_i - Ax_i^v||_2^2 + \lambda_1||A||_2^2$$
$$\min_B \sum_{i=1}^{N}\sum_{v=1}^{V}||q_i^v - Bz_i^v||_2^2 + \lambda_2||B||_2^2$$

which yields the following solutions:

$$\begin{aligned} &A^* = Q\sum_{v=1}^{V}X^{vT}(\sum_{v=1}^{V}X^v X^{vT} + \lambda_1 I)^{-1}, \\ &Q = [q_1, ..., q_N], X = [x_1, ..., x_N], \\ &B^* = \sum_{v=1}^{V}Q^v Z^{vT}(\sum_{v=1}^{V}Z^v Z^{vT} + \lambda_2 I)^{-1}, \\ &Q^v = [q_1^v, ..., q_N^v], Z^v = [z_1^v, ..., z_N^v]. \end{aligned} \tag{8}$$

Algorithm 1 summarizes our approach. The algorithm converged after a few iterations in our experiments.

4. Experiments

We evaluated our approach for both cross-view and multi-view action recognition on the IXMAS multi-view dataset [28]. This dataset contains 11 actions performed three times by ten actors taken from four side views and one top view. Figure 3 shows some example frames. We follow the experiment setting in [20] for extracting the local STIP feature [4]. We first detect up to 200 interest points from each action video and then extract a 100-dimensional gradient-based descriptors around these interest points via PCA. Then these interest points-based descriptors are clustered into 1000 visual words by k-mean clustering and

Algorithm 1 Learning View-invariant Sparse Representations for Cross-view Action Recognition

```
Input: $Y^v = [Y_1^v, ..., Y_K^v], Q, Q^v, v = 1, ..., V, \lambda, \eta$
Initialize $D$ and $D^v$
for $k = 1 \rightarrow K$ do
    Initialize class-specific dictionary $D_k$ in $D$ by solving $D_k = arg\min_{D_k, \alpha_k} ||[Y_k^1 ... Y_k^V] - D_k \alpha_k||_F^2 + \lambda||\alpha_k||_1$
    Initialize class-specific dictionary $D_k^v$ in $D^v$ by solving $D_k^v = arg\min_{D_k^v, \beta_k^v} ||Y_k^v - D_k^v \beta_k^v||_F^2 + \lambda||\beta_k||_1$
end for
repeat
    Compute sparse codes $x_i^v$, $z_i^v$ of a set of correspondence videos $y_i^v$ by solving the multi-task group LASSO problem in (4) using the SLEP [18]
    Update each atom $d_j$ in $D$ and $d_j^v$ in $D^v$ using (6) and (7) respectively
    Update transformation matrices $A$, $B$ using (8)
until convergence or certain rounds
Output: $D = [D_1, ..., D_K], D^v = [D_1^v, ..., D_K^v]$
```

each action video is represented by a 1000-dimensional histogram. For the global feature, we extract shape-flow descriptors introduced in [26] and learn a codebook of size 500 by k-means clustering on these shape-flow descriptors. Similarly, this codebook is used to encode shape-flow descriptors and each action video is represented by a 500-dimensional histogram. Then the local and global feature descriptors are concatenated to form a 1500-dimensional descriptor to represent an action video.

For fair comparison to [6, 20, 15], we use three evaluation modes: (1) *unsupervised correspondence* mode; (2) *supervised correspondence* mode ; (3) *partially labeled* mode. For the first two correspondence mode, we use 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. And all videos of the orphan action are excluded when learning the quantized visual words and constructing dictionaries. The only difference between the first and the second mode is whether the category labels of the correspondence videos are available or not. For the third mode, we follow [15] to consider a semi-supervised setting where a small portion of videos from the target view is labeled and no matched correspondence videos exist. From this we want to show that our framework can be applied to the domain adaptation problem. Two comparing methods for the third mode are two types of SVMs used in [1]. The first one is AUGSVM, which creates a feature-augmented version of each individual feature as the new feature. The second one is MIXSVM which trains two SVM's on the source and target views and learns an optimal linear combination of them.

Note that the test actions from the source and target views are not seen during dictionary learning whereas the test action can be seen in the source view for classifier training in the first two evaluation modes. On the contrary, the test action from different views can be seen during both dictionary learning and classifier training in the third mode. For all modes, we report the classification accuracy by averaging the results over different combinations of selecting orphan actions.

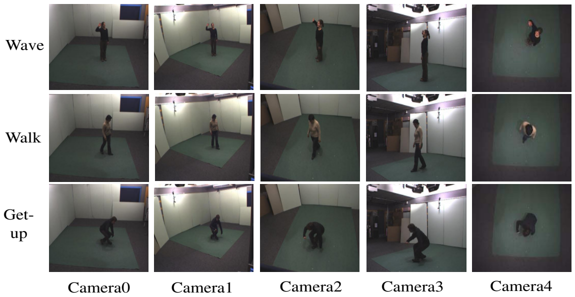
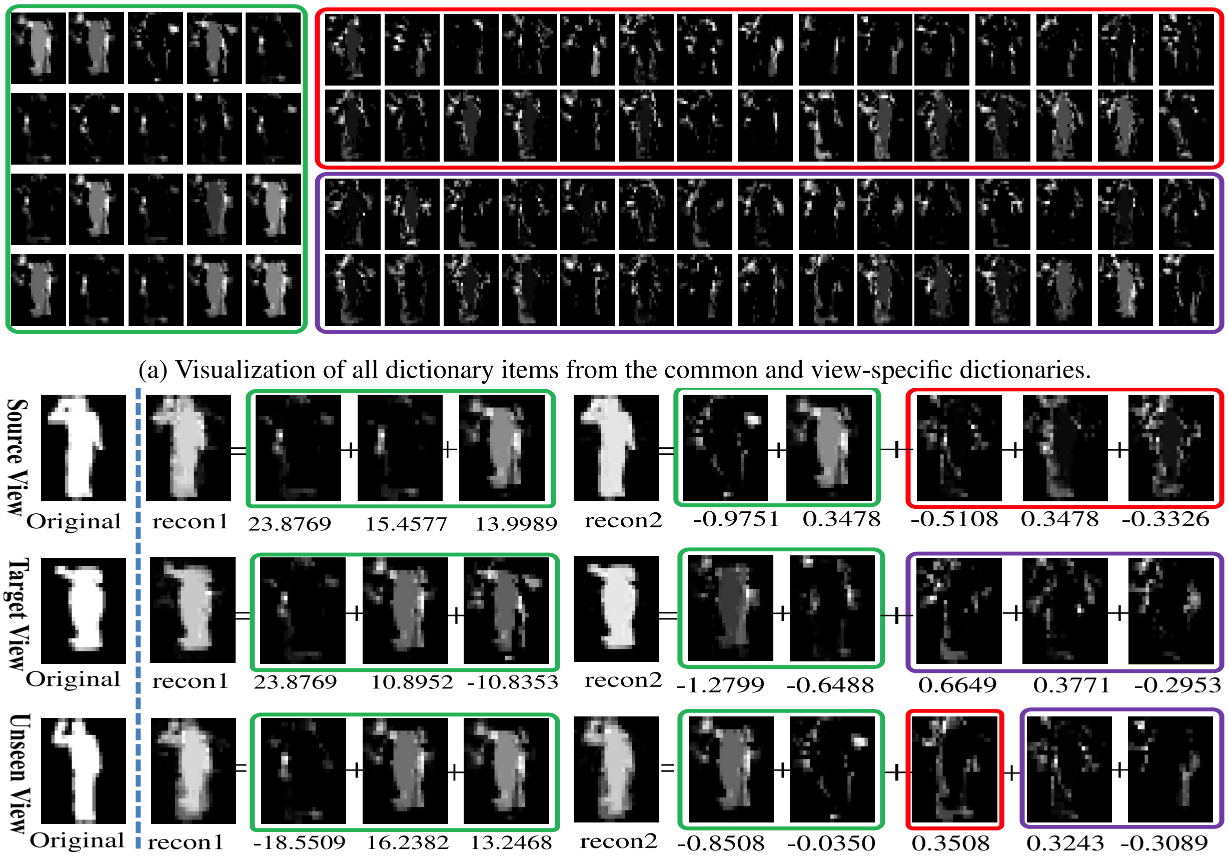


Figure 3. **Exemplar frames from the IXMAS multi-view dataset.** Each row shows one action viewed across different angles.

4.1. Benefits of the Separation of the Common and View-specific Dictionaries

In this section, we demonstrate the benefits of the separation of the common and view-specific dictionaries. For visualization purpose, two action classes ”check-watch” and ”waving” taken by Camera0 and Camera2 from the IXMAS dataset was selected to construct a simple cross-view dataset. We extract the shape descriptor [16] for each video frame and learn a common dictionary and two-view specific dictionaries using our approach. We then reconstruct a pair of frames taken from Camera0 and Camer2 views of the action ”waving” using two methods. The first one is to use the common dictionary only to reconstruct the frame pair. The other one is use both the common dictionary and the view-specific dictionary for reconstruction. Figure 2(b) shows the original shape feature and the reconstructed shape features of two frames of action ”waving” from two seen views and one unseen view using the mentioned two methods. First, comparing dictionary items in D and $\{D^s, D^t\}$, we see that some items in D mainly encode the body and body outline which are just shared by frames of the same action from two view while items in $\{D^s, D^t\}$ mainly encode different arm poses that reflects the class information in the two views. It demonstrates that the common dictionary has the ability to exploit view-shared features from different views. Second, it can be observed that better reconstruction is achieved by using both the common dictionary D and view-specific dictionaries. This is because the common dictionary may not reconstruct the more detailed view-specific features well such as arm poses. The separation of the common dictionary enables the view-specific dictionaries to focus on exploiting and aligning view-specific features from different views. Third, from the last row in Figure 2(b), we find that a good reconstruction of an action frame taken from the unseen view can be achieved by using the common dictionary only. It demonstrates that the common dictionary learned from two seen views has the capability to represent videos of the same action from an unseen view. Moreover, two



(a) Visualization of all dictionary items from the common and view-specific dictionaries.

(b) Reconstruction of shape features of action ”waving” from two seen views and one unseen view.

Figure 2. **Illustration of the benefits of the common dictionary.** (a) Visualization of all dictionary atoms in D (green color), D^s (red color) and D^t (purple color). (b) Figures from $2 \sim 5$ columns show the reconstruction result using D only. Figures from $6 \sim 11$ columns show the reconstruction result using $\{D, D^s\}$, $\{D, D^t\}$ and $\{D, D^s, D^t\}$ respectively. Only at most top-3 dictionary items are shown.

%	C0	C1	C2	C3	C4
C0		(77.6, 79.9, 81.8, **99.1**)	(69.4, 76.8, 88.1, **90.9**)	(70.3, 76.8, 87.5, **88.7**)	(44.8, 74.8, 81.4, **95.5**)
C1	(77.3, 81.2, 87.5, **97.8**)		(73.9, 75.8, 82.0, **91.2**)	(67.3, 78.0, **92.3**, 78.4)	(43.9, 70.4, 74.2, **88.4**)
C2	(66.1, 79.6, 85.3, **99.4**)	(70.6, 76.6, 82.6, **97.6**)		(63.6, 79.8, 82.6, **91.2**)	(53.6, 72.8, 76.5, **100.0**)
C3	(69.4, 73.0, 82.1, **87.6**)	(70.0, 74.4, 81.5, **98.2**)	(63.0, 66.9, 80.2, **99.4**)		(44.2, 66.9, 70.0, **95.4**)
C4	(39.1, 82.0, 78.8, **87.3**)	(38.8, 68.3, 73.8, **87.8**)	(51.8, 74.0, 77.7, **92.1**)	(34.2, 71.1, 78.7, **90.0**)	
Ave.	(63.0, 79.0, 83.4, **93.0**)	(64.3, 74.7, 79.9, **95.6**)	(64.5, 75.2, 82.0, **93.4**)	(58.9, 76.4, 85.3, **87.1**)	(46.6, 71.2, 75.5, **95.1**)

Table 1. **Cross-view action recognition accuracies of different approaches on the IXMAS dataset using *unsupervised correspondence* mode.** Each row corresponds to a source (training) view and each column a target (test) view. The four accuracy numbers in the bracket are the average recognition accuracies of [11], [20], [15] and our unsupervised approach respectively.

%	C0	C1	C2	C3	C4
C0		(79, **98.5**)	(79, **99.7**)	(68, **99.7**)	(76, **99.7**)
C1	(72, **100.0**)		(74, **97.0**)	(70, **89.7**)	(66, **100.0**)
C2	(71, **99.1**)	(82, **99.3**)		(76, **100.0**)	(72, **99.7**)
C3	(75, **90.0**)	(75, **99.7**)	(73, **98.2**)		(76, **96.4**)
C4	(80, **99.7**)	(73, **95.7**)	(73, **100.0**)	(79, **98.5**)	
Ave.	(74, **97.2**)	(77, **98.3**)	(76, **98.7**)	(73, **97.0**)	(72, **98.9**)

Table 2. **Cross-view action recognition accuracies of different approaches on the IXMAS dataset using *supervised correspondence* mode.** Each row corresponds to a source (training) view and each column a target (test) view. The accuracy numbers in the bracket are the average recognition accuracies of [7] and our supervised approach respectively.

%	C0	C1	C2	C3	C4
C0		(42.8, 36.8, 63.6, **64.9**)	(45.2, 46.8, 60.0, **64.1**)	(47.2, 42.7, 61.2, **67.1**)	(30.5, 36.7, 52.6, **65.5**)
C1	(44.1, 39.4, 61.0, **63.6**)		(43.5, 51.8, **62.1**, 60.2)	(47.1, 45.8, 65.1, **66.7**)	(43.6, 40.2, 54.2, **66.8**)
C2	(53.7, 49.1, 63.2, **65.4**)	(50.5, 49.4, 62.4, **63.2**)		(53.5, 45.0, **71.7**, 67.1)	(39.1, 46.9, 58.2, **65.9**)
C3	(46.3, 39.3, 64.2, **65.4**)	(42.5, 42.5, **71.0**, 61.9)	(48.8, 51.2, 64.3, **65.4**)		(37.5, 38.9, 56.6, **61.6**)
C4	(37.0, 40.3, 50.0, **65.8**)	(35.0, 42.5, 59.7, **62.7**)	(44.4, 40.4, 60.7, **64.5**)	(37.2, 40.7, 61.1, **61.9**)	
Ave.	(45.3, 42.6, 59.6, **65.0**)	(42.7, 42.8, **64.2**, 63.2)	(45.4, 47.5, 61.9, **63.5**)	(46.2, 43.5, 64.8, **65.7**)	(37.6, 40.7, 55.4, **65.0**)

Table 3. **Cross-view action recognition accuracies of different approaches on the IXMAS dataset using *partially* labeling mode.** Each row corresponds to a source (training) view and each column a target (test) view. The accuracy numbers in the bracket are the average recognition accuracies of AUGSVM, MIXSVM from [1], [15], and our approach respectively.

methods have nearly the same reconstruction performance for frames of the same action from the unseen view. This is because $\{D^s, D^t\}$ are learned by exploiting features that are specific for the two seen views. In addition, the separation of the common dictionary and view-specific dictionaries can enable us to learn more compact view-specific dictionaries.

4.2. Cross-view Action Recognition

We evaluate our approach using three different modes. We first learn a common dictionary D and two view-specific dictionaries $\{D^s, D^t\}$ corresponding to the source and target views respectively. Both D and D^s are used to represent the training videos in the source view. Similarly, for a test video y in the target view, we encode it over $\hat{D} = [D \;\; D^t]$, *i.e.* $\beta = arg\min_\beta ||y - \hat{D}\beta||_2^2 + \lambda_0||\beta||_1$ where λ_0 is a parameter to balance the reconstruction error and sparsity. For the first two modes, a k-NN classifier is used to classify the test video in the sparse feature space. For the third mode, we use SRC method [30] to predict the label of y, *i.e.* $k^* = arg\min_k ||y - \hat{D}_k\beta_k||_2^2 + \lambda_0||\beta_k||_1$ where $\hat{D}_k = [D_k \;\; D_k^t]$ and β_k is the associated sparse codes.

As shown in Tables 1 and 2, our approach yields a much better performance for all 20 combinations for the first two modes. Moreover, the proposed approach achieves more than 90% recognition accuracy for most combinations. The higher recognition accuracy obtained by our supervised setting over our unsupervised setting demonstrates that the dictionaries learned using labeled information across views are more discriminative.

For the partially labeled mode, our approach outperforms other approaches for most of source-target combinations in Table 3. It is interesting to note that for the case where Camera4 is the source or target view, the recognition accuracies of comparing approaches are a little lower than other combinations of piecewise views. This is because the Camera4 was set above the actors and different actions look very similarly from the top view. However, our approach still achieves a very high recognition accuracy for these combinations, which further demonstrates the effectiveness of our approach.

%	C0	C1	C2	C3	C4	Avg
Ours (mode1)	**97.0**	**99.7**	**97.2**	**98.0**	**97.3**	**97.8**
Ours (mode2)	**99.7**	**99.7**	**98.8**	**99.4**	**99.1**	**99.3**
[33] (mode1)	98.5	99.1	99.1	100	90.3	97.4
[33] (mode2)	99.4	98.8	99.4	99.7	93.6	98.2
[20]	86.6	81.1	80.1	83.6	82.8	82.8
[11]	74.8	74.5	74.8	70.6	61.2	71.2
[19]	76.7	73.3	72.0	73.0	N/A	73.8
[29]	86.7	89.9	86.4	87.6	66.4	83.4

Table 4. **Multi-view action recognition results using the unsupervised and supervised correspondence modes.** Each column corresponds to one target view.

%	C0	C1	C2	C3	C4
Ours (mode3)	**66.6**	**68.4**	**65.4**	67.2	**67.8**
[15]	62.0	65.5	64.5	**69.5**	57.9
AUGSVM	54.2	50.8	58.1	49.5	46.9
MIXSVM	46.4	44.2	52.3	47.7	44.7

Table 5. **Multi-view action recognition results using the partially labeled mode.** Each column corresponds to one target view.

4.3. Multi-view Action Recognition

We select one camera as a target view and use all other four cameras as source views to explore the benefits of combining multiple source views. Here we use the same classification scheme used for cross-view action recognition. Both D and the set of correspondence dictionaries D^v are learned by aligning the sparse representations of shared action videos across all views. Since videos from all views are aligned into a common view-invariant sparse feature space, we do not need to differentiate the training videos from each source view in this common view-invariant sparse feature space.

Table 4 shows the average accuracy of the proposed approach for the first two evaluation modes. Note that the comparing approaches are evaluated using the unsupervised correspondence mode. Both our unsupervised and supervised approaches outperform other comparing approaches and achieve nearly perfect performance for all target views. Furthermore, [20, 33] and our unsupervised approach only use training videos from four source views to train a classifier while other approaches used all the training videos from all five views to train the classifier. Table 5 shows the average accuracy of different approaches using the *partially labeled* evaluation mode. The proposed approach outperforms [15] on four out of five target views. Overall, we accomplish a comparable accuracy with [15] under the *partially labeled* mode.

5. Conclusion

We presented a novel dictionary learning framework to learn view-invariant sparse representations for cross-view action recognition. We propose to simultaneously learn a common dictionary to model view-shared features and a set of view-specific dictionaries to align view-specific features from different views. Both the common dictionary and the corresponding view-specific dictionary are used to represent videos from each view. We transfer the indices of non-zeros in the sparse codes of videos from the source view to the sparse codes of the corresponding videos from the target view. In this way, the mapping between the source and target views is encoded in the common dictionary and view-specific dictionaries. Meanwhile, the associated sparse representations are view-invariant because non-zero positions in the sparse codes of correspondence videos share the same set of indices. Our approach can be applied to cross-view and multi-view action recognition under unsupervised, supervised and domain adaptation settings.

References

[1] L. T. Alessandro Bergamo. Exploiting weakly-labeled web images to improve object classification: a domain adaptation approach. In *NIPS*, 2010. 5, 7

[2] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. In *ICCV*, 2005. 1

[3] G. K. M. Cheung, S. Baker, and T. Kanade. Shape-from-silhouette of articulated objects and its use for human body kinematics estimation and motion capture. In *CVPR*, 2003. 1

[4] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *VS-PETS*. 4

[5] A. A. Efros, A. C. Berg, G. Mori, and J. Malik. Recognizing action at a distance. In *ICCV*, 2003. 1

[6] A. Farhadi and M. K. Tabrizi. Learning to recognize activities from the wrong view point. In *ECCV*, 2008. 1, 2, 5

[7] A. Farhadi, M. K. Tabrizi, I. Endres, and D. A. Forsyth. A latent model of discriminative aspect. In *ICCV*, 2009. 1, 6

[8] C.-H. Huang, Y.-R. Yeh, and Y.-C. F. Wang. Recognizing actions across cameras by exploring the correlated subspace. In *ECCV Workshops*, 2012. 1, 2

[9] Z. Jiang, Z. Lin, and L. S. Davis. Learning a discriminative dictionary for sparse coding via label consistent k-svd. In *CVPR*, 2011. 3

[10] I. N. Junejo, E. Dexter, I. Laptev, and P. Pérez. View-independent action recognition from temporal self-similarities. *IEEE Trans. Pattern Anal. Mach. Intell.* 2

[11] I. N. Junejo, E. Dexter, I. Laptev, and P. Pérez. Cross-view action recognition from temporal self-similarities. In *ECCV*, 2008. 2, 6, 7

[12] S. Kong and D. Wang. A dictionary learning approach for classification: Separating the particularity and the commonality. In *ECCV*, 2012. 4

[13] I. Laptev and T. Lindeberg. Space-time interest points. In *ICCV*, 2003. 1

[14] B. Li, O. I. Camps, and M. Sznaier. Cross-view activity recognition using hankelets. In *CVPR*, 2012. 2

[15] R. Li and T. Zickler. Discriminative virtual views for cross-view action recognition. In *CVPR*, 2012. 2, 5, 6, 7, 8

[16] Z. Lin, Z. Jiang, and L. S. Davis. Recognizing actions by shape-motion prototype trees. In *ICCV*, 2009. 1, 5

[17] J. Little and J. E. Boyd. Recognizing people by their gait: The shape of motion. *Videre*, 1996. 1

[18] J. Liu, S. Ji, and J. Ye. *SLEP: Sparse Learning with Efficient Projections*. Arizona State University, 2009. 4, 5

[19] J. Liu and M. Shah. Learning human actions via information maximization. In *CVPR*, 2008. 1, 7

[20] J. Liu, M. Shah, B. Kuipers, and S. Savarese. Cross-view action recognition via view knowledge transfer. In *CVPR*, 2011. 1, 2, 4, 5, 6, 7

[21] F. Lv and R. Nevatia. Single view human action recognition using key pose matching and viterbi path searching. In *CVPR*, 2007. 1

[22] V. Parameswaran and R. Chellappa. Human action-recognition using mutual invariants. *CVIU*, 2005. 2

[23] V. Parameswaran and R. Chellappa. View invariance for human action recognition. *IJCV*, 2006. 2

[24] D.-S. Pham and S. Venkatesh. Joint learning and dictionary construction for pattern recognition. In *CVPR*, 2008. 4

[25] C. Rao, A. Yilmaz, and M. Shah. View-invariant representation and recognition of actions. *IJCV*, 2002. 2

[26] D. Tran and A. Sorokin. Human activity recognition with metric learning. In *ECCV*, 2008. 5

[27] A. ul Haq, I. Gondal, and M. Murshed. On dynamic scene geometry for view-invariant action matching. In *CVPR*, 2011. 2

[28] D. Weinland, E. Boyer, and R. Ronfard. Action recognition from arbitrary views using 3d exemplars. In *ICCV*, 2007. 2, 4

[29] D. Weinland, M. Özuysal, and P. Fua. Making action recognition robust to occlusions and viewpoint changes. In *ECCV*, 2010. 2, 7

[30] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2009. 3, 7

[31] P. Yan, S. M. Khan, and M. Shah. Learning 4d action feature models for arbitrary view action recognition. In *CVPR*, 2008. 2

[32] A. Yilmaz and M. Shah. Actions sketch: A novel action representation. In *CVPR*, 2005. 1

[33] J. Zheng, Z. Jiang, J. Phillips, and R. Chellappa. Cross-view action recognition via a transferable dictionary pair. In *BMVC*, 2012. 1, 2, 7