A Group Sparsity-Driven Approach to 3-D Action Recognition

Serhan Coşar Müjdat Çetin
Faculty of Engineering and Natural Sciences
Sabancı University, 34956 İstanbul, TURKEY
{serhancosar,mcetin}@sabanciuniv.edu

Abstract

In this paper, a novel 3-D action recognition method based on sparse representation is presented. Silhouette images from multiple cameras are combined to obtain motion history volumes (MHVs). Cylindrical Fourier transform of MHVs is used as action descriptors. We assume that a test sample has a sparse representation in the space of training samples. We cast the action classification problem as an optimization problem and classify actions using group sparsity based on l_1 regularization. We show experimental results using the IXMAS multi-view database and demonstrate the superiority of our method, especially when observations are low resolution, occluded, and noisy and when the feature dimension is reduced.

1. Introduction

With its vast amount of application areas, including, surveillance, human-computer interaction, and virtual reality, action recognition became a popular field in the last decade. The majority of work in this field focuses on using a single camera. However, using a single camera suffers from problems such as dependency on viewpoint and problems due to self-occlusion. Therefore, multi-camera systems are becoming more preferable over single camera systems.

One of the earliest works based on a single camera is the work of Bobick and Davis [1]. They construct motion templates by aggregating the differences between subsequent silhouettes. Two parallel studies extend these motion templates to three dimensions, and call them Motion History Volumes (MHVs) [15, 4]. MHV is a single volumetric data cube constructed using silhouettes from multiple cameras and it encodes the dynamics of the motion in 3-D space. In [15], MHV is transformed into cylindrical coordinates to provide view invariance and Fourier analysis is used for feature extraction. In [4], Canton-Ferrer et al. use 3-D invariant Hu moments to provide view invariance. In addition, Pehlivan et al. [12] utilize visual hulls, constructed using silhouettes from multiple cameras, for action recognition. They encode layers of a visual hull by circles and extract features such as number of circles, area of outer circles, etc. from each layer.

Compressed sensing and sparse representation (SR) have become important signal recovery techniques because of their success in various application areas [7, 11, 13, 14]. In some applications, sparse representation has also been used as a classification method. Assuming that a test sample can be represented as a sparse linear combination of training samples, l_1 regularization is used to find this linear combination and, thereby, find the identity of the test sample [16, 17]. In [16], sparse representation is used for face recognition. Classification of human motion using wearable motion sensor signals is performed via sparse representation in [17]. Furthermore, this line of thought has also been used for the single camera action recognition problem [10, 8, 9]. In [10], a combination of 2-D silhouettes and optical flow is used as features. The covariance matrix of bag of optical flow features is used in [8]. MoSIFT features are extracted and used in [9].

In this paper, we propose a multi-camera action recognition method that is based on sparse representation. We represent the 3-D motion by constructing MHVs using the silhouettes from multiple cameras. To describe the ongoing action, cylindrical Fourier transform of MHVs are used as features [15]. As in [16, 17], we assume that a test sample can be written as a linear combination of training samples from the class it belongs. In other words, a test sample has a sparse representation in the space covered by the training samples. In particular, our assumption is that a test sample can be represented accurately by the group of training samples from the right class, whereas contributions from other training samples would be minor. Based on this assumption, we cast the classification problem as an optimization problem and solve it by enforcing group sparsity through l_1 regularization.

The rest of the paper is organized as follows. In Section 2 we briefly explain motion history volumes and the action descriptors extracted from them. Then in Section 3 we give the details of our classification method based on sparse representation. Experimental results under various conditions are presented in Section 4. Finally, we draw conclusion in Section 5.

2. MHVs and Action Descriptors

2.1. Construction of MHVs

Motion history volumes are extensions of 2-D motion templates, first introduced by Bobick and Davis in [1], to 3-D [15]. They represent the dynamics of the motion in 3-D.

At each camera, after acquiring the images, silhouettes are extracted. The silhouette images obtained from multiple cameras are used to create visual hulls at each time instance. By using these visual hulls an occupancy function, $D(x, y, z, t)$, that represents the presence of a person in space and time, is defined. $D(x, y, z, t)$, is set to 1 if the point (x, y, z) is 1 in the visual hull created at time t, and set to 0 otherwise. By using this occupancy function, a motion history volume is constructed as follows:

$$v_\tau(x,y,z,t) = \begin{cases} \tau & \text{if } D(x,y,z,t)=1 \\ max(0, v_\tau(x,y,z,t-1)-1) & o.w. \end{cases} \tag{1}$$

where τ is the maximum duration of the motion at point (x, y, z) [15].

With respect to the duration of an action, the volumes found by Eq. 1 are normalized and final motion history volumes are obtained:

$$v(x,y,z) = \frac{v_{\tau=t_{max}-t_{min}}(x,y,z,t_{max})}{t_{max}-t_{min}} \tag{2}$$

where t_{min} and t_{max} are start and end time of an action. As in [15], t_{min} and t_{max} are estimated by searching for the local minima in the global motion energy of MHVs. An example of MHV constructed for “kicking“ action is shown in Figure 1.

2.2. Action Descriptor Extraction

To be able to recognize actions robustly, a method that is invariant to rotation, scale and translation is needed. But, since MHVs encode space occupancy, they are not invariant. Because of the nature of human motions, it is reasonable to assume that similar actions only differ by rigid transformations composed of scale, translation, and rotation around the z-axis [15]. As in [15], we express MHVs in a cylindrical coordinate-system:

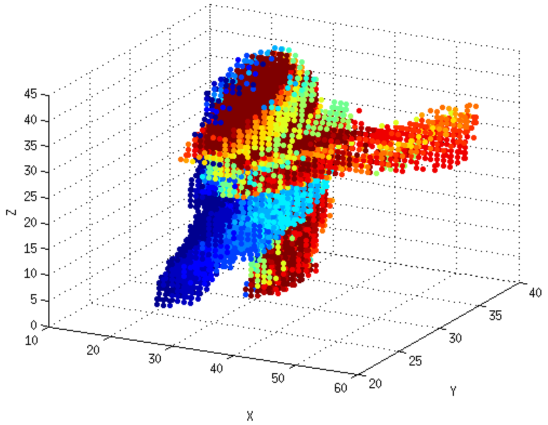


Figure 1. An example of MHV constructed for “kicking“ action. Color indicates the values of MHV.

$$v(\sqrt{x^2+y^2}, tan^{-1}\left(\frac{y}{x}\right), z) \rightarrow v(r,\theta,z) \tag{3}$$

Thus rotations around the z-axis results in cyclical translation shifts:

$$v(x\cdot cos\theta_0 + y\cdot sin\theta_0, -x\cdot sin\theta_0 + y\cdot cos\theta_0, z) \rightarrow v(r, \theta+\theta_0, z) \tag{4}$$

The absolute values of 1-D Fourier transform along the θ dimension for each value of r and z, $|V(r, k_\theta, z)|$, are used as motion descriptors:

$$V(r,k_\theta,z) = \int_{-\pi}^{\pi} v(r,\theta,z)e^{-j2\pi k_\theta \theta} d\theta \tag{5}$$

Motion descriptor extraction for ”kicking” is illustrated in Figure 2.

By the *shift property* of Fourier transform, a shift in the θ dimension corresponds to phase modulation in frequency domain. As a result, 1-D Fourier magnitudes are invariant to rotation along θ. Before taking the Fourier transform, the location and scale dependencies of MHVs are removed by centering around the center of mass, and scale normalization. Therefore, the motion descriptors obtained by this procedure are invariant to rotation, scale and translation. We use these descriptors as features in our method.

3. Classification using Sparse Representation

In the feature space, we assume that each action class satisfies a low-dimensional subspace model. If a valid test sample can be represented as a linear combination of all training samples, the dominant coefficients in the sparsest representation correspond to the training samples from

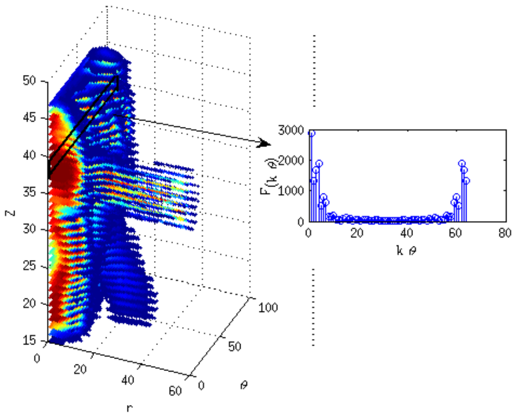


Figure 2. Action descriptors are constructed by taking Fourier transform over θ for couples of values (r, z) in cylindrical coordinates and concatenating the Fourier magnitudes.

the underlying action class, and hence they indicate the membership of the test sample.

Mathematically, we express this as follows: in a feature space of m dimensions, given n_i training samples of the ith action class, there is a relation between a test sample, $y \in \mathrm{R}^m$, and the training samples, $\{v_{i,j}\}_{j=1}^{n_i}$, from the same class:

$$y = \alpha_{i,1} v_{i,1} + \alpha_{i,2} v_{i,2} + \cdots + \alpha_{i,n_i} v_{i,n_i} \tag{6}$$

where $\alpha_{i,j} \in \mathrm{R}, j = 1, 2, \cdots, n_i$.

Writing the training samples of ith class as the columns of a matrix, we obtain the matrix $\psi_i = [v_{i,1}, v_{i,2}, \cdots, v_{i,n_i}] \in \mathrm{R}^{m \times n_i}$. Since we do not know the class of the test sample initially, by concatenating the n training samples of all k object classes, we obtain the following matrix ψ:

$$\psi \doteq [\psi_1, \psi_2, \cdots, \psi_k] \in \mathrm{R}^{m \times n} \qquad n = \Sigma_{i=1}^{k} n_i \tag{7}$$

As in [16, 17], by rewriting the relation in Eq. 6 using the matrix ψ, we obtain the following linear representation of y in terms of all training samples:

$$y = \psi x \in \mathrm{R}^m \tag{8}$$

where $x = [0, \cdots, 0, \alpha_{i,1}, \alpha_{i,2}, \cdots, \alpha_{i,n_i}, 0, \cdots, 0]^T \in \mathrm{R}^n$ is a coefficient vector, all of whose entries except those corresponding to the ith class are zero. Thus, solving the system in Eq. 8 for x gives the identity of the test sample y. In practice, since real data are noisy, it may not be possible to express the test sample exactly as a superposition of the training samples. In other words, a noise term can be added to the linear system in Eq. 8:

$$y = \psi x + z \tag{9}$$

where $z \in \mathrm{R}^m$ is a noise term with bounded energy $||z||_2 < \epsilon$.

For a large number of object classes, this representation is naturally *group sparse* – meaning that there are non-zero coefficients corresponding to a particular group (class) of training samples and zeros elsewhere. Different from the procedure in [16, 17], we define a vector, x', as follows:

$$x' = [x'_1, x'_2, \cdots, x'_k] \in \mathrm{R}^k \qquad x'_i = ||\{\alpha_{i,j}\}_{j=1}^{n_i}||_2 \tag{10}$$

Namely, the ith element of the vector x' is the l_2 norm of the coefficients corresponding to training samples of the ith class. Since the vector x is group sparse, x' is a sparse vector. Therefore, to classify the test sample, we are interested in finding the group sparse solution to $y = \psi x + z$ by solving the following optimization problem:

$$\hat{x} = \arg\min_x ||x'||_0 \text{ subject to } ||y - \psi x||_2 < \epsilon \tag{11}$$

where $||.||_0$ denotes the l_0 norm and counts the number of nonzero entries in a vector. However, the problem of minimizing the l_0 norm is NP-hard. Recent development in the emerging theory of compressed sensing [6, 3, 5] reveals that if the vector x' is sparse enough, the solution of the l_0-minimization problem in Eq. 11 is equal to the solution to the following l_1-minimization problem:

$$\hat{x} = \arg\min_x ||x'||_1 \text{ subject to } ||y - \psi x||_2 < \epsilon \tag{12}$$

In fact, we optimize the Lagrangian form of this problem:

$$\hat{x} = \arg\min_x ||y - \psi x||_2 + \lambda ||x'||_1 \tag{13}$$

where λ is the regularization parameter. This convex optimization problem can be efficiently solved via second-order cone programming [2]

After finding the group sparse representation $\hat{x}$ of the test sample y, we classify it based on how well the coefficients associated with all training samples of each action class reproduce y. For each class i, let $\delta_i : \mathrm{R}^n \rightarrow \mathrm{R}^n$ be the characteristic function that selects the coefficients associated with the ith class. For $x \in \mathrm{R}^n$, $\delta_i(x) \in \mathrm{R}^n$ is a new vector whose only nonzero entries are the entries in x that are associated with class i. Using only the coefficients associated with the ith class, one can approximate the given test sample y as $\tilde{y} = \psi \delta_i(\hat{x}_i)$. We then classify y based on these approximations by assigning it to the object class that minimizes the residual between y and $\tilde{y}$:

$$\hat{i} = \arg\min_i r_i(y) \doteq \arg\min_i ||y - \psi \delta_i(\hat{x}_i)||_2 \tag{14}$$

In [16, 17], it is assumed that the test sample can be represented by a small number of training samples from the same class and hence the vector x is considered as sparse (rather than group sparse). A similar formulation could be obtained in our setting by replacing x' by x in Eqs. 12 and 13. After finding the sparse representation, the test sample can be classified again by using Eq. 14. In Section 4, we also present the results of this sparsity-based approach and compare it to the group sparsity based approach described above.

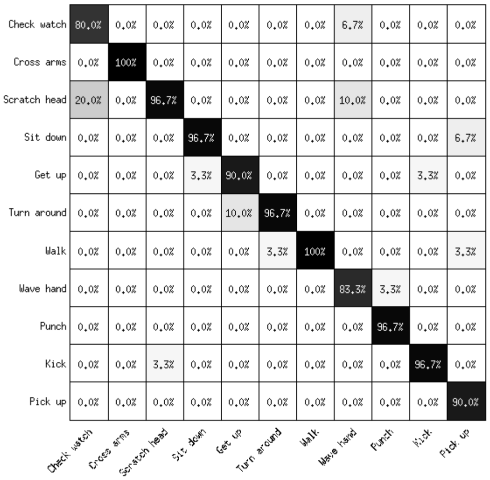
4. Experimental Results

We test our method on the publicly available IXMAS dataset [15], which is a popular dataset used for evaluating multi-view action recognition methods. The dataset consists of 11 actions (check watch, cross arms, scratch head, sit down, get up, turn around, walk, wave, punch, kick, point) and each action is performed three times with free orientation and position by 10 different actors. Actions are recorded by five synchronized and calibrated cameras. Example views from the dataset are shown in Figure 3. We use the visual hulls provided with the dataset on a 64x64x64 voxel grid. While constructing the MHVs, the motion segmentation method in [15] is used. Action descriptors are created on a 32x32x32 voxel grid. The classification results presented here are based on leave-one-out cross-validation, i.e., we train on data from 9 actors and test on the remaining actor; repeat this for all combinations of actors; and average the results.

We have compared our method with the method in [15]. In [15], first principal component analysis (PCA) is applied for dimensionality reduction and then, three different procedures are performed to classify action descriptors: 1) a new action is classified according to the Euclidean distance to class means; 2) a new action is classified according to the Mahalanobis distance to class means; 3) Fisher linear discriminant analysis (LDA) is performed to maximize the between-class scatter and minimize the within-class scatter, then a new action is classified according to the Euclidean distance to class means.

	The method in [15]			**SR**	
Action	LDA	PCA	Maha.	Group Sparse	Sparse
Check watch	83.33%	46.66%	**86.66%**	80.00%	83.33%
Cross arms	**100.00%**	83.33%	**100.00%**	**100.00%**	**100.00%**
Scratch head	93.33%	46.66%	93.33%	**96.66%**	**96.66%**
Sit down	93.33%	93.33%	93.33%	**96.66%**	**96.66%**
Get up	**90.00%**	83.33%	93.33%	**90.00%**	**90.00%**
Turn around	**96.66%**	93.33%	**96.66%**	**96.66%**	**96.66%**
Walk	**100.00%**	**100.00%**	**100.00%**	**100.00%**	**100.00%**
Wave hand	**90.00%**	53.33%	80.00%	83.33%	86.66%
Punch	93.33%	53.33%	96.66%	**96.66%**	**96.66%**
Kick	93.33%	83.33%	**96.66%**	**96.66%**	**96.66%**
Pick up	83.33%	66.66%	**90.00%**	**90.00%**	**90.00%**
Average	92.42%	66.36%	93.33%	93.33%	**93.93%**

Table 1. Accuracies of the method in [15] and our SR based method for each action. Bold values represent the best accuracy for each action.



	Check watch	Cross arms	Scratch head	Sit down	Get up	Turn around	Walk	Wave hand	Punch	Kick	Pick up
Check watch	80.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	6.7%	0.0%	0.0%	0.0%
Cross arms	0.0%	100%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Scratch head	20.0%	0.0%	96.7%	0.0%	0.0%	0.0%	0.0%	10.0%	0.0%	0.0%	0.0%
Sit down	0.0%	0.0%	0.0%	96.7%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	6.7%
Get up	0.0%	0.0%	0.0%	3.3%	90.0%	0.0%	0.0%	0.0%	0.0%	3.3%	0.0%
Turn around	0.0%	0.0%	0.0%	0.0%	10.0%	96.7%	0.0%	0.0%	0.0%	0.0%	0.0%
Walk	0.0%	0.0%	0.0%	0.0%	0.0%	3.3%	100%	0.0%	0.0%	0.0%	3.3%
Wave hand	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	83.3%	3.3%	0.0%	0.0%
Punch	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	96.7%	0.0%	0.0%
Kick	0.0%	0.0%	3.3%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	96.7%	0.0%
Pick up	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	90.0%

Figure 4. Confusion matrix of our method based on group sparsity. Columns and rows represent the true and assigned classes respectively.

Table 1 presents the performance of our method and the method in [15] for each action and averaged over all actions. For the framework proposed in this paper, we presented the results of both the *group sparse* approach based on Eq. 13 as well as the *sparse* approach described at the very end of Section 3. We have empirically set the regularization parameter in Eq. 13, λ, to 500 and 100 for the group sparse and sparse approaches, respectively. The proposed framework is run under Matlab on a Pentium Core2Quad 2.83Ghz computer and the processing time of the non-optimized code for a single test sample is approximately 23.7 s for group sparse and 30.5 s for sparse approaches, respectively. For the results of [15] in Table 1, the column titled ”PCA” corresponds to the Euclidean distance-based approach in [15] and the other two columns correspond to the LDA and Mahalanobis distance-based versions. It can be seen that in three actions our method achieves a better level of accuracy than the method in [15]. In six actions, our method and the method in [15] achieves the same results. Just for ”check watch” and ”wave hand” actions, the method in [15] achieves better results. In the average, group sparse version of our method achieves the best level of accuracy. Figure 4 shows the confusion matrix of our method based on group sparsity. We observe that with the exception of ”check watch” and ”wave hand” actions both versions of our approach achieve the same level of accuracy. For ”check watch” and ”wave hand” actions sparse version performs better than the group sparse version.

We have also performed tests under various conditions. In the next subsections, the results of experiments when action descriptors are low resolution, when data

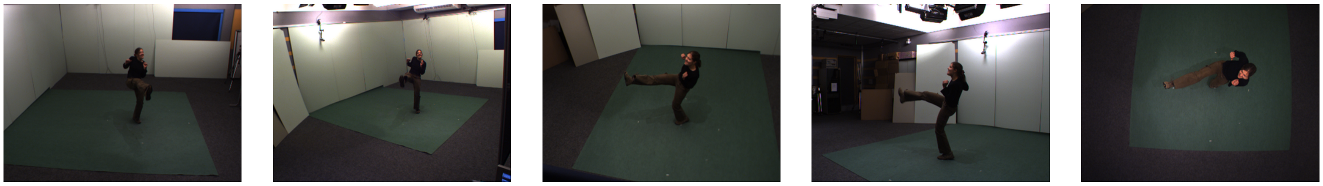


Figure 3. Example views from the IXMAS dataset recorded by five synchronized and calibrated cameras [15].

	The method in [15]			SR	
y	LDA	PCA	Maha.	Group Sparse	Sparse
[32]	92.42%	66.36%	93.33%	93.33%	**93.93%**
[16]	78.18%	44.55%	77.27%	90.00%	**90.61%**
[8]	20.91%	9.09%	9.09%	**73.64%**	67.88%

Table 2. Average accuracies of the method in [15] and our SR based method when action descriptors are low-resolution. Bold values represent the best accuracy for each row.

are noisy, when there is occlusion, and when the feature dimension is reduced are presented.

4.1. Low Resolution Data

In this experiment, action descriptors created in lower voxel grid sizes are used to test the robustness of our method in the case of resolution loss. Our method and the method in [15] are tested by using action descriptors in 16x16x16 and 8x8x8 voxel grid sizes. The average accuracies obtained in these experiments together with the average accuracies obtained using the 32x32x32 action descriptors are presented in Table 2.

These results show that even when the action descriptors have very low resolution our method achieves reasonable level of accuracy. For the 16x16x16 grid size, the performance of the method in [15] degrades much more dramatically than that of our method. While the method in [15] achieves reasonable level of accuracy (78.18%), our method achieves better accuracy level (90.00%). When the action descriptors have a resolution of 8x8x8, the method in [15] exhibits an unacceptable level of accuracy. PCA and Mahalanobis procedures achieve only random assignment accuracies (9.09%), whereas our method achieves a reasonable level of accuracy (73.64%). These experiments demonstrate the robustness and superiority of our proposed approach in the case of low resolution data.

In Table 2, we have also presented the results of the *sparse* version of our approach. When the action descriptors have a resolution of 16x16x16, sparse version performs slightly better than the group sparse version (90.61%). But for the resolution of 8x8x8, the sparse version achieves a lower level of accuracy than the group sparse version (67.88%).

		The method in [15]			SR	
y	Corruption	LDA	PCA	Maha.	Group Sparse	Sparse
[32]	0%	92.42%	66.36%	93.33%	93.33%	**93.93%**
[32]	10%	89.39%	42.42%	90.30%	92.73%	**93.33%**
[32]	20%	65.15%	16.06%	63.03%	**93.03%**	92.73%
[32]	30%	25.76%	9.09%	24.55%	**91.52%**	90.61%
[32]	40%	11.52%	9.09%	13.33%	**92.42%**	**92.42%**
[32]	50%	11.21%	9.09%	12.42%	**90.91%**	90.30%
[32]	60%	10.00%	9.09%	10.30%	89.39%	**90.30%**
[32]	70%	10.61%	9.09%	10.00%	**86.67%**	86.36%
[32]	80%	9.70%	9.09%	9.39%	**83.94%**	**83.94%**
[32]	90%	9.70%	9.09%	9.09%	**81.82%**	80.91%
[32]	100%	9.39%	9.09%	9.09%	**84.55%**	83.94%

Table 3. Average accuracies of the method in [15] and our method on data corrupted by zero-mean Gaussian noise with variance specified in terms of percentages of the maximum value of the MHV. Bold values represent the best accuracy for each row.

4.2. Corrupted Data

Poor performance in temporal segmentation affects the values of the MHVs. If the start and/or end times of the motion is miscalculated, the values of MHVs will be inaccurate (Eq. 2). To test the robustness of our method for such perturbations, we have corrupted the MHVs with zero-mean Gaussian noise. The test sample is created by extracting the action descriptors from these corrupted MHVs. Training samples are created by using the original MHVs. We have performed experiments with various noise variances which are specified in terms of percentages of the maximum values of the MHVs.

Average accuracies achieved by the method in [15] and our method for various noise levels are presented in Table 3 and Figure 5. The results obtained from the original data (0% corruption) are also shown for comparison. It can be observed that our method outperforms the method in [15] for all noise variances considered in this experiment. The lowest accuracy achieved by our method is when the noise variance is 90% of the maximum of the MHV and it is a reasonable rate (81.82%). For all variances, we also observe that group sparse and sparse versions of our approach achieve similar results. On the other hand, there is a significant performance drop for the method in [15] as the data become more noisy (most notably when the noise variance increases from 20% to 30%). For the tests in which the noise variance is selected equal to or greater than 60% of the maximum of the MHV, the best accuracy obtained by the method in [15] is close to random

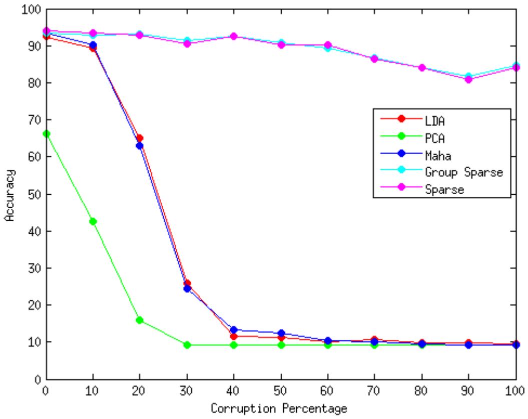


Figure 5. Accuracies of the method in [15] and our method on data corrupted by zero-mean Gaussian noise with variance specified in terms of percentages of the maximum value of the MHV.

assignment accuracy (9.09%). These results show that our method is robust to reductions in data quality, which may be the result of, e.g., failures in temporal segmentation.

4.3. Occluded Data

Occlusion is one of the most common and important problems in real world scenarios. Occlusion occurs most commonly through an object occluding parts of the person in the scene. In addition, background subtraction methods that generate the silhouettes may not produce accurate segmentations. In our problem both of these scenarios would lead to inaccurate silhouettes and can be treated as occlusion problems. Since the visual hulls are constructed by using these occluded silhouettes, they will also be occluded and, consequently, MHVs will be occluded as well.

In this experiment, we have examined how the occlusion in MHVs affects the recognition accuracy. Starting from the center of the MHV, we have occluded (set to zero) the MHVs in various levels, from 5 percent to 90 percent. Test samples are created by extracting the action descriptors from these occluded MHVs and training samples are created by using the original MHVs. Due to the steps involved in feature extraction, occlusion of MHVs has a non-trivial effect in the feature space [15]. In particular, this effect involves all feature components rather than being limited to occlusion of a subset of the feature components. Given this non-trivial effect, in all of our experiments with all techniques, we assume the presence of a perfect occlusion detector for the sake of simplicity. The occluded points have not been taken into account in feature extraction steps of both our method and the method in [15]. In practice, a real occlusion detection algorithm needs to be used within all of these methods.

		The method in [15]			**SR**	
y	Occlusion	LDA	PCA	Maha.	Group Sparse	Sparse
[32]	0%	92.42%	66.36%	93.33%	93.33%	**93.93%**
[32]	5%	91.21%	66.97%	90.91%	91.21%	**91.52%**
[32]	10%	88.18%	65.15%	88.18%	89.70%	**90.61%**
[32]	20%	89.09%	64.24%	89.39%	**91.52%**	90.91%
[32]	30%	86.06%	64.24%	88.18%	**90.00%**	89.39%
[32]	40%	86.36%	62.12%	86.67%	87.27%	**87.88%**
[32]	50%	85.15%	61.52%	83.33%	**85.76%**	85.45%
[32]	60%	83.33%	57.88%	82.42%	**85.45%**	84.55%
[32]	70%	75.76%	59.39%	75.15%	79.39%	**80.61%**
[32]	80%	70.30%	60.61%	70.61%	**76.06%**	73.03%
[32]	90%	47.88%	46.36%	47.58%	**56.67%**	53.94%

Table 4. Average accuracies of the method in [15] and our method for various levels of occlusion. Bold values represent the best accuracy for each row.

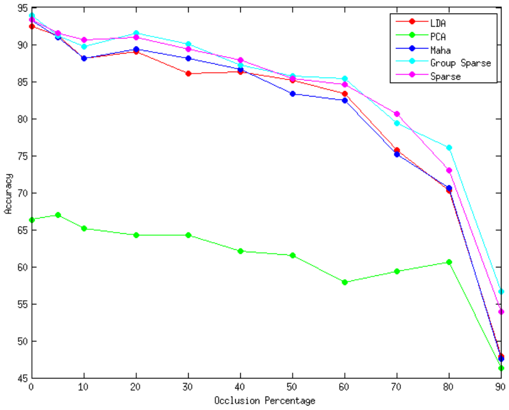


Figure 6. The plot of accuracies of the method in [15] and our method for various levels of occlusion.

In Table 4 and Figure 6, the average accuracies obtained by the method in [15] and our method for various levels of occlusion are presented. The results obtained from the original data (0% occlusion) are also presented for comparison. The results show that our method performs better than the method in [15] for all levels of occlusion. For occlusion levels up to 60%, the accuracies of the method in [15] are close to the accuracies of our method. But, for higher occlusion levels, our method is definitely better than the method in [15].

The accuracies of the *sparse* version are also presented in Table 4 and Figure 6. Comparing the results of the group sparse and sparse versions, it can be observed that the two versions achieve similar accuracy levels.

4.4. Reduced Feature Dimension

In this experiment, we have tested the robustness of our method when the feature space has lower dimension, e.g., because of missing features or simple dimensionality

		The method in [15]			**SR**	
y	Dim.(m)	LDA	PCA	Maha.	Group Sparse	Sparse
[32]	16,384	92.42%	66.36%	93.33%	93.33%	**93.93%**
[32]	15,565	92.12%	66.36%	90.91%	92.42%	**92.73%**
[32]	14,746	91.52%	64.55%	90.61%	90.61%	**92.12%**
[32]	13,926	90.61%	64.24%	90.61%	90.30%	**91.21%**
[32]	13,107	86.67%	60.91%	86.97%	87.58%	**87.88%**
[32]	12,288	82.12%	58.18%	83.33%	84.24%	**84.55%**
[32]	11,469	74.85%	55.15%	73.33%	**77.58%**	76.67%
[32]	10,650	66.06%	51.52%	65.15%	69.70%	**69.39%**
[32]	9,830	57.27%	39.70%	50.00%	63.03%	**66.06%**

Table 5. Average accuracies of the method in [15] and our method for various feature dimensions. Bold values represent the best accuracy for each row.

reduction. For various feature dimensions, our method and the method in [15] are tested. The average accuracies obtained in these experiments together with the average accuracies obtained using the original features are shown in Table 5.

We observe that our approach (either the *sparse* or the *group sparse* version) exhibits better performance than the method in [15].

5. Conclusion

A novel multi-camera action recognition method based on sparse representation has been proposed in this paper. MHVs constructed using the silhouettes from multiple cameras have been used to represent the motion dynamics in 3-D space. As in [15], cylindrical Fourier transform of MHVs are used to describe the action in the scene. Following the work in [16, 17] from other recognition [16] or sensing [17] contexts, we assume that the feature vector of a test sample can be sparsely represented by feature vectors of the training set. We develop two parallel perspectives one based on regular sparsity and the other one based on so called group sparsity. Then in this framework, the action classification problem is cast as an optimization problem and l_1 regularization is used for its solution. We have presented the results of our method in various conditions including low-resolution data, occlusion, noise, and low-dimensional features. We have also compared our results with the results of the method in [15] and shown performance improvements it provides, especially in the case of limitations in data quality and quantity. In such cases, the sparsity constraint imposed in our framework helps us achieve better level of accuracy.

In addition, we have observed that group sparse and sparse approaches achieve similar results on average. The sparse approach is based on the idea that a test sample can be represented by a small number of training samples, regardless of the class labels of the training samples. On the other hand, the group sparse approach imposes more structure, imposing sparsity across classes (i.e., allowing only a small number of classes to be active in the representation) while allowing the use of a large number of training samples from the active classes. In different problems and/or feature spaces, there can be cases where one or the other performs better.

The classification framework presented in this paper is a generic framework that can be used together with different features than those considered in this paper, including those that do not involve MHVs. It may also be applied to different action recognition problems based on different features. Each of the scenes we used in our experiments contained a single person. When scenes with multiple humans are considered, data association issues emerge. Extending the ideas presented in this paper to such scenarios could be an interesting line of future work.

Acknowledgements

This work was partially supported by a Turkish Academy of Sciences Distinguished Young Scientist Award and by a graduate fellowship from the Scientific and Technological Research Council of Turkey.

References

[1] A. Bobick and J. Davis. The recognition of human movement using temporal templates. *IEEE Transactions On Pattern Analysis And Machine Intelligence*, 23(3):257–267, Mar 2001. 1, 2

[2] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, Mar. 2004. 3

[3] E. Candes, J. Romberg, and T. Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics*, 59(8):1207–1223, Aug 2006. 3

[4] C. Canton-Ferrer, J. R. Casas, and M. Pardàs. Human model and motion based 3d action recognition in multiple view scenarios (invited paper). In *14th European Signal Processing Conference, EUSIPCO*, 2006. ISBN: 0-387-34223-0. 1

[5] D. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, Apr 2006. 3

[6] D. Donoho. For most large underdetermined systems of linear equations the minimal l(1)-norm solution is also the sparsest solution. *Communications on Pure and Applied Mathematics*, 59(6):797–829, Jun 2006. 3

[7] M. Duarte, M. Davenport, D. Takhar, J. Laska, T. Sun, K. Kelly, and R. Baraniuk. Single-pixel imaging via compressive sampling. *IEEE Signal Processing Magazine*, 25(2):83 –91, March 2008. 1

[8] K. Guo, P. Ishwar, and J. Konrad. Action Recognition Using Sparse Representation on Covariance Manifolds of Optical Flow. In *7th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 2010. 1

[9] A. Liu and D. Han. Spatiotemporal Sparsity Induced Similarity Measure for Human Action Recognition. *International Journal of Digital Content Technology and its Applications*, 4(8):143–149, 2010. 1

[10] C. Liu, Y. Yang, and Y. Chen. Human action recognition using sparse representation. In *IEEE International Conference on Intelligent Computing and Intelligent Systems (ICIS), 2009*, 2009. 1

[11] J. Mairal, M. Elad, and G. Sapiro. Sparse representation for color image restoration. *IEEE Transactions on Image Processing*, 17(1):53 –69, Jan. 2008. 1

[12] S. Pehlivan and P. Duygulu. A new pose-based representation for recognizing actions from multiple cameras. *Computer Vision Image Understanding*, 115(2), Feb 2011. 1

[13] M. Plumbley, T. Blumensath, L. Daudet, R. Gribonval, and M. Davies. Sparse representations in audio and music: From coding to source separation. *Proceedings of the IEEE*, 98(6):995 –1005, June 2010. 1

[14] L. Potter, E. Ertin, J. Parker, and M. Cetin. Sparsity and compressed sensing in radar imaging. *Proceedings of the IEEE*, 98(6):1006 –1020, June 2010. 1

[15] D. Weinland, R. Ronfard, and E. Boyer. Free viewpoint action recognition using motion history volumes. *Computer Vision Image Understanding*, 104:249–257, November 2006. 1, 2, 4, 5, 6, 7

[16] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma. Robust Face Recognition via Sparse Representation. *IEEE Transactions On Pattern Analysis And Machine Intelligence*, 31(2):210–227, June 2009. 1, 3, 7

[17] A. Y. Yang, R. Jafari, S. S. Sastry, and R. Bajcsy. Distributed Recognition of Human Actions Using Wearable Motion Sensor Networks. *Journal of Ambient Intelligence and Smart Environments*, pages 1–5, 2009. 1, 3, 7