Modeling and Recognizing Action Contexts in Persons Using Sparse Representation

Kai-Ting Chuang, Jun-Wei Hsieh*, and Yilin Yan

Depart. of Computer Science and Engineering, National Taiwan Ocean University,
No.2, Beining Rd., Keelung 202, Taiwan
shieh@mail.ntou.edu.tw

Abstract. This paper proposes a novel dynamic sparse representation-based classification scheme to treat the problem of interaction action analysis between persons using sparse representation. The occlusion problem and the difficulty to model complicated interactions are the major challenges in person-to-person action analysis. To address the occlusion problem, the proposed scheme represents an action sample in an over-complete dictionary whose base elements are the training samples themselves. This representation is naturally sparse and makes errors (caused by different environmental changes like lighting or occlusions) sparsely appear in the training library. Because of the sparsity, it is robust to occlusions and lighting changes. The difficulty of complicated action modeling can be tackled by adding more examples to the over-complete dictionary. Thus, even though the interaction relations are complicated, the proposed method still works successfully to recognize them and can be easily extended to analyze action events among multiple persons.

Keywords: Sparse Coding, Sparse Representation, Occlusions, daily events.

1 Introduction

Human action analysis [1]- [18]] is an important task in various application domains like video surveillance [1], video retrieval [18], human-computer interaction systems, and so on. Characterization of human action is equivalent to dealing with a sequence of video frames that contain both spatial and temporal information. The challenge in human action analysis is how to properly characterize spatial-temporal information and then facilitate subsequent comparison/recognition tasks. To treat this challenge, some approaches build various action syntactic primitives to represent and recognize events. For example, in [18], Park and Aggarwal used the "blob" concept to model and segment a human body into different body parts from which human events were analyzed using the dynamic Bayesian networks. Wang, Huang, and Tan [6] used the R transform to extract contour features from different frames and then proposed a HMM-based recognition scheme to analyze human behaviors. Some approaches decompose actions into sequences of key atomic action units which are referred to as atoms. For example, in [12], Gaidon, Harchaoui, Schmid proposed an atom

* Corresponding author.

J.-S. Pan et al. (Eds.): *Advances in Intelligent Systems & Applications,* SIST 21, pp. 531–541.
DOI: 10.1007/978-3-642-35473-1_53

sequence model (ASM) to represent the temporal structure of actions and then recognize actions in videos using a sequence of "atoms" which are obtained by manual annotations. In addition to videos, humans can recognize activities based on only still images. However, the prerequisite that body parts or poses must be well estimated makes this scheme inappropriate for real-time analysis of human behaviors.

To avoid the difficulty of action primitive or body part extraction, some approaches extract feature points of interest and obtain their motion flows to represent and recognize actions. For example, Rosales and Sclaroff [13] proposed a trajectory-based recognition system to detect pedestrians in outdoor environments and then recognize their activities from multiple views using mixtures of Gaussian classifiers. In [9], Laptev *et al.* generated the concept of interest points (STIP) from images to flow volumes and then used it to extract various key frames to represent action events. The success of the above feature-flow methods strongly depends on a large set of well tracking points to describe action changes across frames.

In addition to event features, another key problem in event analysis is how to model the temporal and spatial dynamics of events. To treat this problem, Mahajan *et al.* [8] proposed a layer concept to divide the recognition task to three layers, *i.e.*, physical, logical, and event layers corresponding to feature extraction, action representation, and event analysis, respectively. Hidden Markov model (HMM) is another commonly used scheme to model event dynamics. For example, Messing, Pal, and Kautz [10] tracked a set of corners to obtain their velocity histories and then used HMM to learn activity event models. The challenges related to HMMs involve how to specify and learn the HMM model structure.

A particular action between two objects can vary significantly under different conditions such as camera views, person's clothing, object appearances, and so on. Thus, it is more challenging to analyze human events happening between two objects because of their complicated interaction relations. In addition, occlusions between objects often happen and lead to the failure of action recognition. In [17], Filipovych and Ribeiro used a probabilistic graphical model to recognize the primitive actor-object interaction events like "grasping" or "touching" a fork (or a spoon, a cup). In addition to videos, some previous works address joint modeling of human poses, objects and relations among them from still images. For example, in [15], Yao and Fei-Fei proposed a random field model to encode the mutual connections of components in the analyzed object, the human pose, and the body parts to recognize human-object interaction activities in still images. However, until now, reliable estimation of body configurations for persons in any poses remains a very challenging problem.

This paper addresses the problem of action analysis between persons (or human-object interactions) using sparse representation. Fig. 1 shows the flowchart of our system. As described before, the complicated interaction changes and the occlusion problem between two objects increase many challenges in action recognition. To the above problems, this paper proposes a novel dynamic sparse representation scheme to represent an event in an over-complete dictionary whose base elements are the training samples themselves. If sufficient training samples are collected from each action class, each test sample will be possibly represented as a linear combination of just the training sample from the same class. This representation is naturally sparse, involving only a small fraction of the overall training database. The sparse property also makes errors (caused by different environmental changes like lighting or occlusions) sparsely appear in the training library as a special case of training samples to be handled. The sparsity of error distribution increases the robustness of our scheme to occlusion. After that, a sparse reconstruction cost (SRC) is is proposed to classify action events to more categories. Even though the interaction relations are complicated, the proposed method still works successfully to recognize them and can be easily extended to analyze action events among multiple persons.

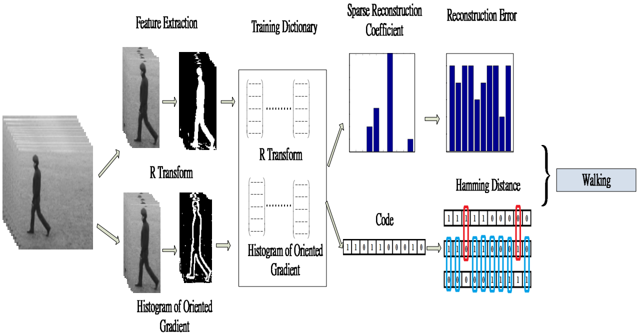


Fig. 1. Flowchart of the proposed system

2 Feature Extraction

This paper uses two features to represent an object shape, *i.e.*, *R* transform and HOG. Details of these two features are described as follows.

2.1 Radon Transform and R Transform

Radon transform in two dimensions is the integral transform consisting of the integral of an image over straight line. Let $I(x, y)$ denote one input image and L be a straight line with the equation: $x\cos\theta + y\sin\theta = t$. Then, the Radon transform of $I(x, y)$ along L is defined as follows:

$$Radon\{I\} = P(\theta, t) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} I(x, y)\delta(x\cos\theta + y\sin\theta - t)dxdy\,, \qquad (1)$$

where $\theta \in [0,\pi]$ and $t \in (-\infty,\infty)$ 、 $\delta(x)$ is a Dirac delta function. In Eq.(1), the result of Radon Transform is a two dimensional signal. It can be converted to 1D signal by the improved form of Radon transform [**6**]:

$$R(\theta) = \int_{-\infty}^{\infty} P^2(\theta, t)dt\,. \qquad (2)$$

Eq.(2) is the R transform of $I(x, y)$. If $I(x, y)$ is normalized to a fixed size, the R transform is invariant under translation and scaling.

2.2 Histogram of Oriented Gradients

In addition to R transformation, this paper also uses the histogram of oriented gradients (HOG) [11] to describe a posture. Let I_x and I_y denote the central differences at point (x, y) are given by

$$I_x = I(x+1, y) - I(x-1, y) \text{ and } I_y = I(x, y+1) - I(x, y-1), \tag{3}$$

where $I(x, y)$ is the intensity value of the point (x, y). Then, the gradient magnitude $M(x, y)$ and its orientation $\theta(x, y)$ can be computed by

$$M(x, y) = \sqrt{I_x^2 + I_y^2} \text{ and } \theta(x, y) = \tan^{-1} I_x / I_y . \tag{4}$$

For a given grid, we can then construct a HOG descriptor with 8 bin, where each bin accumulates the number of edge points whose angles $\theta(x, y)$ fall in this angle bin. Then, through various combining among different grids, an ensemble of HOG descriptors can be formed for action analysis.

3 Sparse Representation

Sparse representation [3]-[5] is a technique to build an overcomplete dictionary to represent a target. In this paper, we utilize the sparse representation and dictionary learning techniques to design a novel framework to analyze action events happening between multiple persons.

3.1 Dictionary Initialization and Matching Pursuit

Let x denote an n-dimensional feature vector, i.e., $x \in R^n$. We say that it admits a sparse approximation over a *dictionary* D in $R^{n \times K}$, where each column vector is referred to as an *atom.* Consider a finite training set of signals $X = [x_1, x_2, ..., x_N] \in R^{n \times N}$. Then, one can find a linear combination of a "few" atoms from D that is "close" to the signal x; that is,

$$X \approx DA, \tag{5}$$

where $A = [\alpha_1, ..., \alpha_N] \in R^{K \times N}$ is the set of combination coefficients in the sparse decomposition. Given X, the K-SVD algorithm maintains the best representation of each signal with strict sparsity constraints to learn the over complete dictionary D. It is an iterative scheme alternating between sparse coding of the training signals with respect to the current dictionary and an update process for the dictionary atoms so as to better fit the training signals. Then, the learning process can be formulated a joint optimization problem with respect to the dictionary D and A of the sparse decomposition as

$$\arg\min_{D,A} \| X - DA \|_2^2 \quad s.t. \forall i, \ \| \alpha_i \|_0 \leq T, \tag{6}$$

where $D = [d_1, ..., d_K] \in R^{n \times K}$, $A = [\alpha_1, ..., \alpha_N] \in R^{K \times N}$, T is the most desired number of non-zero coefficients, and $\| \alpha_i \|_0$ is the l_0-norm which counts the number of nonzeros in a vector α_i. Eq.(6) can be formulated as another equivalent problem:

$$\arg\min_{D,\alpha_i} \| \alpha_i \|_0 \quad s.t. \| x_i - D\alpha_i \|_2^2 \leq \varepsilon \,, \tag{7}$$

where ε is an error tolerance of reconstruction. One of common methods to solve α_i is the Orthogonal Matching Pursuit (OMP). OMP is a greedy method to iteratively solve the optimal α_i for each x_i while fixing D, i.e.,

$$\alpha_i = \min_{A} \| x_i - DA \|_2^2 \;\; s.t. \| \alpha_i \|_0 \leq T \,. \tag{8}$$

At each stage, the OMP selects an atom α_i from D that best resembles the residual.

3.2 Dictionary Learning via K-SVD

This paper applies the K-SVD to solve Eq.(8) through an iterative way with two stages, i.e., sparse coding stage and dictionary update stage. It optimizes **D** and **X** through a number of training iterations until convergence. Each iteration consists of a *sparse coding stage* that optimizes the coefficients in A and a *dictionary update stage* that improves the atoms in **D**. During the *sparse coding stage*, **D** is held while each α_i is optimized by solving Eq.(8) via the OMP scheme, and allowing ach coefficient vector to have no more than T nonzero elements. During the dictionary update stage, each column d_k in D is updated sequentially so that its coefficients can better represent X. The update process is the key inside of K-SVD which accelerates the optimization process of Eq.(6) while maintaining the sparsity requirement. Let d_k denote the kth column in D to be updated. In addition, we denote the coefficients that correspond to d_k, the kth row in A by α_k^{row}. Then, the cost function in Eq.(6) can be rewritten as follows:

$$\begin{aligned}\| X - DA \|_F^2 &= \| X - \sum_{j=1}^{K} d_j \alpha_j^{row} \|_F^2 = \| X - \sum_{j \neq k} d_j \alpha_j^{row} - d_k \alpha_k^{row} \|_F^2 \\ &= \| E_k - d_k \alpha_k^{row} \|_F^2 \,.\end{aligned}$$

The updated values of d_k and α_k^{row} can be obtained by solving

$$\arg\min_{d_k, \alpha_k^{row}} \| E_k - d_k \alpha_k^{row} \|_F^2 \,. \tag{9}$$

The K-SVD scheme suggests the use of the SVD to find alternative d_k and α_k^{row}. If the SVD of E_k is expressed as $\mathrm{USV}^{\mathrm{T}}$, d_k is updated by the first output basis vector u_1 and the non-zero values in α_k^{row} are adjusted to the product of the first singular value $S(1,1)$ and the first column of V.

4 Person-to-Person Action Recognition Using Sparse Representation

This section will deal with different challenges in human action analysis between persons, that is, how to properly characterize spatial-temporal information and how to perform the subsequent comparison/recognition tasks.

4.1 Action Event Representation

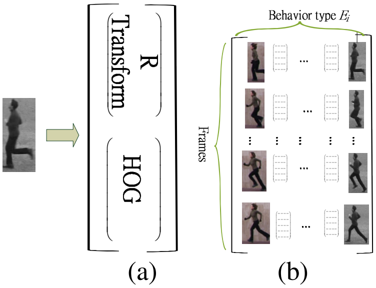


Fig. 2. Sparse representation for analyzing single-person action events. (a) Feature vector to represent an object at each frame. (b) Matrix to represent single person action.

To characterize spatial-temporal information of an action event, this paper uses the R-transform and HOG descriptors to describe each frame. For the R-transform, the angle ranges from 0° to 180° and is further sampled with 4°. As to the HOG descriptor, the observed object is divided to $n_{grid} \times n_{grid}$ grids. Let h_R^X be a vector with 45 elements to represent the R-transform of an object X and h_{hog}^X denote another vector with $8n_{grid}^2$ elements to represent the HOG descriptor of X. Then, as shown in Fig.2 (a), a new feature vector $F^X = (h_R^X, h_{hog}^X)^T$ can be formed to represent X. Let $\oplus$ denote a vector concatenation operation between F^X and F^Y, i.e.,

$$F^X \oplus F^Y = (h_R^X, h_{hog}^X, h_R^Y, h_{hog}^Y)^T . \tag{10}$$

In addition, we use X_t to denote the version of X observed at the tth frame and the superscript k in X^k denote the kth object. If an action event A_X recorded with n_f frames is performed by X, a new feature vector can be formed to represent A through a vector concatenation operation. That is,

$$A_X = F^{X_1} \oplus ... \oplus F^{X_{n_f}} = (h_R^{X_1}, h_{hog}^{X_1}, ..., h_R^{X_{n_f}}, h_{hog}^{X_{n_f}})^T \quad . \tag{11}$$

Let $n = n_f(45 + 8 * n_g^2)$. Then, $A_X \in R^n$ and denotes a normal feature. If an action type E_i is represented by K_i codes, we can use a matrix to represent E_i by the form:

$$E_i = (A_{X^1}, ..., A_{X^k}, ..., A_{X^{K_i}}), \tag{12}$$

where X^k denotes the kth object in E_i and $E_i \in R^{n \times K_i}$. Assume that there are L action types to be recognized and let $K = \sum_{i=1}^{L} K_i$. Then, the basis D in sparse representation for action recognition can be constructed by

$$D = (E_1, \ldots, E_L)\,, \tag{13}$$

where $D \in R^{n\times K}$.

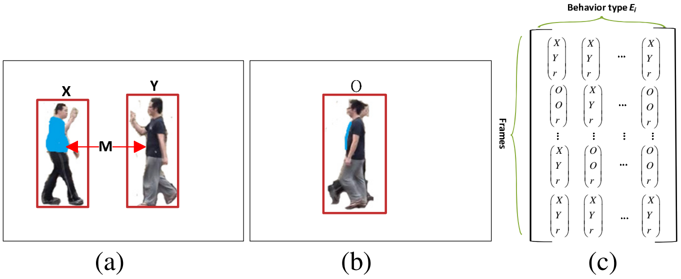


Fig. 3. Sparse representation for analyzing two-person action events. (a) Non-occluded frame. (b) Occluded frame. (c) Matrix to represent single person action event on the same type.

Different from the task of single-person event analysis, there will be different occlusion conditions happening when analyzing two-person interaction events. Assume the interaction events to be analyzed are performed by two persons X and Y. The visual descriptor to capture their spatial relations will change according to the condition whether X and Y are occluded. As Fig.3 (a), if X and Y are not occluded, a new feature descriptor F^{XY} is extracted for describing X and Y as follows:

$$F^{XY} = F^{X} \oplus F^{Y} \oplus (m) = (h_R^X, h_{hog}^X, h_R^Y, h_{hog}^Y, m)^T\,, \tag{14}$$

where m is the motion feature between X and Y. This paper sets m to the relative distance between X and Y. On the other hand, if X and Y are occluded together (see Fig.3 (b)), we replace X and Y with their occluded version O. Then, the descriptor to represent X and Y is constructed as follows:

$$F^{XY} = (h_R^O, h_{hog}^O, h_R^O, h_{hog}^O, m)^T\,, \tag{15}$$

where h_R^O is the R transform of O, h_{hog}^O is the HOG descriptor of O, and m is set to zero. Let A_{XY} denote the action event performed by two persons X and Y. If n_f frames are collected to represent A_{XY}, it can be constructed with the following sparse representation:

$$A_{XY} = F^{XY_1} \oplus \ldots \oplus F^{XY_t} \oplus \ldots \oplus F^{XY_{n_f}}\,. \tag{16}$$

As shown in Fig.3 (c), each column shows the structure of A_{XY}. In the two-person case, if an action type E_i is represented by K_i codes, we can use a matrix to represent E_i by the form:

$$E_i = (A_{XY^1}, \ldots, A_{XY^{K_i}})\,, \tag{17}$$

where X^k denotes the kth object, $E_i \in R^{n\times K_i}$. If there are L action types to be recognized, the library D in sparse representation for action recognition can be formed:

$$D = (E_1,...,E_L)\,, \tag{18}$$

where $D \in R^{n\times K}$ and $K = \sum_{i=1}^{L} K_i$. Then, the K-SVD scheme can be applied to learn the optimal dictionary to more effectively and accurately analyze person-to-person action events.

4.2 Event Classification and Analysis Using Sparse Coding

After obtaining the dictionary D by the K-SVD algorithm, we formulate the action event classification as a signal reconstruction problem. Given an input signal $x \in R^n$, we consider x as a linear combination of column vectors in D , i.e.,

$$x = \alpha_1 d_1 + ... + ... + \alpha_K d_K\,,$$

where $d_k \in D$. Let $\alpha = (\alpha_1,...,\alpha_K)$. The sparse solution α can be obtained by solving the following minimization problem:

$$\arg\min_{\alpha} \| \alpha \|_1 \quad s.t. \| x - D\alpha \|_2^2 \leq \varepsilon\,.$$

This optimization problem can be efficiently solved via the second-order cone programming. Since there are L action types to be recognized, in Eq.(18), the library D is separated to L classes, i.e., $D = (E_1,...,E_L)$. Let $\delta_i : R^n \rightarrow R^n$ be a function that selects the coefficients associated with the ith class. Then, using only the coefficients associated with class i , we compute the residual $r_i(x)$ between x and the approximated one:

$$r_i(x) = \| x - D\delta_i(\alpha) \|_2\,. \tag{19}$$

We can classify x to its corresponding action type by assigning it to the event class that minimizes the residual, i.e.,

$$Type(x) = \arg\min_{i} r_i(x)\,. \tag{20}$$

5 Experimental Results

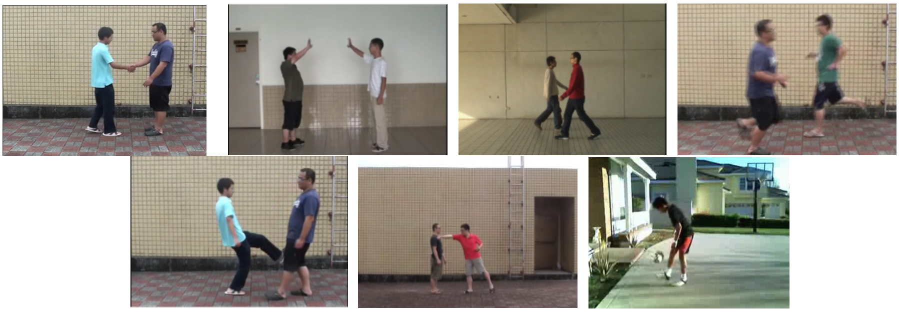
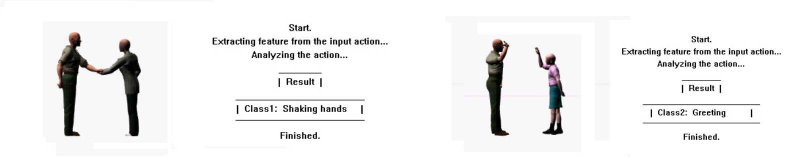


Fig. 4. Examples of real data for seven action types

To evaluate the performance of our proposed method, a real-time system to analyze different action events between two persons at different lighting conditions was implemented. There is no benchmarking database designed for evaluating algorithms to recognize two-person action events. Thus, two kinds of datasets were adopted in this paper for examining the effectiveness of our method, *i.e.*, synthetic and real videos. In this dataset, four kinds of action types were created, *i.e.*, waving, handshaking, running, and walking. For each action type, there were one hundred of action videos created for training and testing, where fifty videos were for training and another set of fifty videos were for testing. In addition to the four types, three extra action types were added in the real dataset for evaluating the effectiveness of our methods under real conditions. The three types are kinking, punching, and soccer-juggling, respectively. The dimension of video frame is 320×240 pixel elements.

Fig. 4 shows the examples of real data for the seven action types. Fig. 5 shows the results of two-person action recognition on the synthetic dataset. All the action types were correctly recognized. Table 1 shows the confusion matrix of two-person action recognition on the synthetic dataset using the SRC method. In this table, the "walking" action type is easily misclassified to the "running" type. The two action types are very similar except their speeds. The "handshaking" action type was sometimes misclassified to the "walking" type since their visual features are similar before handshaking. The average accuracy of the SRC method is about 86%.



```
Start.
Extracting feature from the input action...
Analyzing the action...

| Result |

| Class1:  Shaking hands    |

Finished.
```

```
Start.
Extracting feature from the input action...
Analyzing the action...

| Result |

| Class2:  Greeting       |

Finished.
```

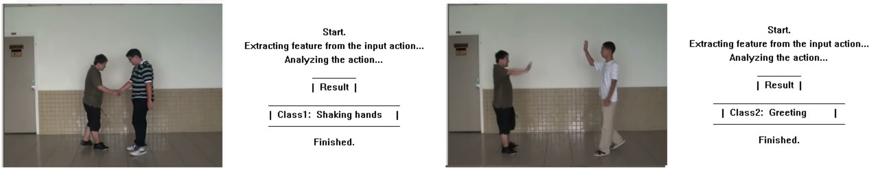
Fig. 5. Results of action recognition on the synthetic dataset

Table 1. Confusion matrix of the SRC method on synthetic dataset

SRC				
Action Types	Handshaking	Greeting	Walking	Running
Handshaking	47/94%	0/0%	0/0%	3/6%
Greeting	0/0%	50/100%	0/0%	0/0%
Walking	8/16%	0/0%	40/80%	2/4%
Running	0/0%	0/0%	15/30%	35/70%

Table 2. Accuracy improvements of the SRC method after adding the speed feature

Action Types methods	Handshaking	Greeting	Walking	Running	Average
SRC	94%	100%	86%	78%	90%

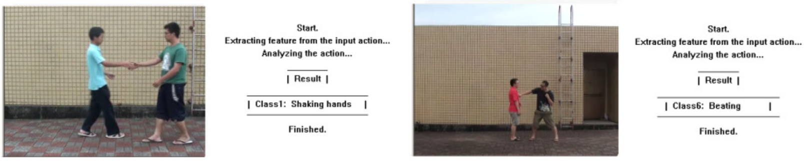


```
Start.
Extracting feature from the input action...
Analyzing the action...

| Result |

| Class1: Shaking hands |

Finished.
```

```
Start.
Extracting feature from the input action...
Analyzing the action...

| Result |

| Class2: Greeting |

Finished.
```

Fig. 6. Results of action recognition on real dataset in indoor environments

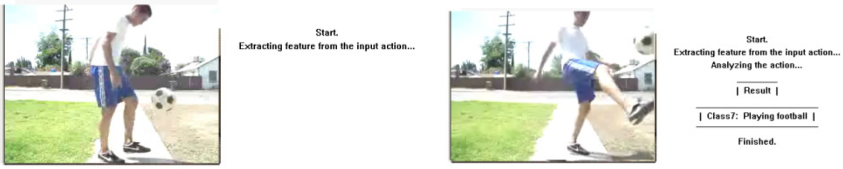
As to the real dataset, seven action types were recognized. The first six types focus on person-to-person action recognition. Fig. 5 shows the results of action type recognition in indoor environments. Fig. 6 shows the results of action type recognition in outdoor environments. Actually, our method can also be applied to recognize person-to-object action events. Thus, the last case is to recognize action events happening between an object (soccer) and a person. Fig. 7 shows the result of recognizing a person-to-object action event. Table 3 shows the confusion matrix of the SRC method on real data. The average accuracy of the SRC method is 80.54%. All the above experiments have proved that the proposed method is a robust, accurate, and powerful tool for action event analysis.



```
Start.
Extracting feature from the input action...
Analyzing the action...

| Result |

| Class1: Shaking hands |

Finished.
```

```
Start.
Extracting feature from the input action...
Analyzing the action...

| Result |

| Class6: Beating |

Finished.
```

Fig. 7. Results of action recognition on real dataset in outdoor environments



```
Start.
Extracting feature from the input action...
```

```
Start.
Extracting feature from the input action...
Analyzing the action...

| Result |

| Class7: Playing football |

Finished.
```

Fig. 8. Result of recognizing a person-to-object event

Table 3. Confusion matrix of the SRC method on real dataset

	SRC						
Action Types	Handshaking	Greeting	Walking	Running	Kicking	Punching	S-juggling
Handshaking	26/81.25%	0/0%	1/3.25%	0/0%	2/6.25%	3/9.25%	0/0%
Greeting	0/0%	30/93.75%	0/0%	0/0%	0/0%	2/6.25%	0/0%
Walking	2/6.25%	0/0%	25/78.125%	5/15.625	0/0%	0/0%	0/0%
Running	0/0%	0/0%	8/31.25%	24/75%	0/0%	0/0%	0/0%
Kicking	0/0%	0/0%	0/0%	0/0%	27/84.375%	5/15.625%	0/0%
Punching	2/6.25%	0/0%	0/0%	0/0%	4/12.5%	26/81.25%	0/0%
Soccer-juggling	1/3.4%	2/6.9%	2/6.9%	2/6.9%	1/3.4%	1/3.4%	20/69%

References

1. Weinland, D., Ronfard, R., Boyer, E.: A survey of vision-based methods for action representation, segmentation, and recognition. Computer Vision and Image Understanding 115(2), 224–241 (2011)
2. Poppe, R.: A survey on vision-based human action recognition. Image and Vision Computing 28(6), 976–990 (2010)
3. Aharon, M., Elad, M., Bruckstein, A.: K-svd: An algorithm for designing overcomplete-dictionries for sparse representation. IEEE Trans. on Signal Processing, 4311–4322 (2006)
4. Qiu, Q., Jiang, Z., Chellappa, R.: Sparse Dictionary-based Representation and Recognition of Action Attributes. In: IEEE Conference on Computer Vision (2011)
5. Mairal, J., Bach, F., Ponce, J., Sapiro, G.: Online learning for matrix factorization and sparse coding. J. Mach. Learn. Res. 11, 19–60 (2010)
6. Wang, Y., Huang, K., Tan, T.: Human activity recognition based on R transform. In: IEEE Conference on Computer Vision and Pattern Recognition (2007)
7. Kratz, L., Nishino, K.: Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1446–1453 (2009)
8. Mahajan, D., Kwatra, N., Jain, S., Kalra, P.: A framework for activity recognition and detection of unusual activities. In: International Conference on Graphic and Image Processing (2004)
9. Laptev, I., Perez, P.: Retrieving actions in movies. In: International Conference on Computer Vision (October 2007)
10. Messing, R., Pal, C., Kautz, H.: Activity recognition using the velocity histories of tracked keypoints. In: International Conference on Computer Vision (October 2009)
11. Dalal, N., Triggs, B.: Histograms of Oriented Gradients for Human Detection. In: IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 886–893 (2005)
12. Gaidon, A., Harchaoui, Z., Schmid, C.: Actom Sequence Models for Efficient Action Detection. In: IEEE Conference on Computer Vision and Pattern Recognition (2011)
13. Rosales, R., Sclaroff, S.: 3D Trajectory Recovery for Tracking Multiple Objects and Trajectory Guided Recognition of Actions. In: Proc. of IEEE Conf. on Computer Vision and Pattern Recognition, vol. 2, pp. 117–123 (1999)
14. Nguyen, N.T., Bui, H.H., Venkatesh, S., West, G.: Recognition and monitoring high-level behaviours in complex spatial environments. In: IEEE International Conference on Computer Vision and Pattern Recognition, Madison, Wisconsin, USA, vol. 2, pp. 620–625 (June 2003)
15. Yao, B., Fei-Fei, L.: Recognizing Human-Object Interactions in Still Images by Modeling the Mutual Context of Objects and Human Poses. To appear in IEEE Transactions on Pattern Analysis and Machine Intelligence
16. Delaitre, V., Sivic, J., Laptev, I.: Learning person-object interactions for action recognition in still images. Advances in Neural Information Processing Systems (2011)
17. Filipovych, R., Ribeiro, E.: Recognizing Primitive Interactions by Exploring Actor-Object States. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–7 (June 2008)
18. Park, S., Park, J., Aggarwal, J.K.: Video Retrieval of Human Interactions Using Model-based Motion Tracking and Multi-Layer Finite State Automata. In: Bakker, E.M., Lew, M., Huang, T.S., Sebe, N., Zhou, X.S. (eds.) CIVR 2003. LNCS, vol. 2728, pp. 394–403. Springer, Heidelberg (2003)