Discriminative Key Pose Extraction using Extended LC-KSVD for Action Recognition

Lijuan Zhou*, Wanqing Li†, Yuyao Zhang*, Philip Ogunbona†, Duc Thanh Nguyen* and Hanling Zhang‡
University of Wollongong, Wollongong, NSW, Australia, 2522*†
Hunan University, Changsha, P. R. China, 410012‡
Email: {lz683, yz606, dtn156}@uowmail.edu.au*, {wanqing, philip}@uow.edu.au†, jt_hlzhang@hnu.edu.cn‡

***Abstract*—This paper presents a method for extracting discriminative key poses for skeleton-based action recognition. Poses are represented by normalized joint locations, velocities and accelerations of skeleton joints. An extended label consistent K-SVD (ELC-KSVD) algorithm is proposed for learning the common and action-specific dictionaries. Discriminative key poses are represented by the atoms of the action-specific dictionaries. With the specific dictionaries, sparse codes are obtained for representing action instances through max pooling and temporal pyramid. A SVM classifier is trained for action recognition. The proposed method was evaluated on the MSRC-12 gesture and MSR-Action 3D datasets. Experimental results have shown that the proposed method is effective in extracting discriminative key poses.**

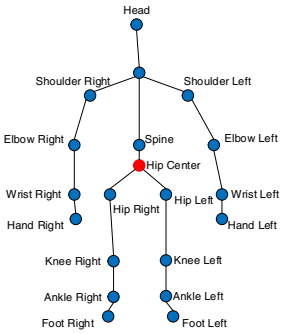

Fig. 1. Skeleton joints.

I. Introduction

Recognizing human action is one of the most active research topics in computer vision with a variety of applications. For example, in the video-based surveillance, action recognition is used to detect abnormal activities in order to ensure public safety. In education and health care, it can be applied for real-time monitoring of the elderly people, children and patients; and for movement disorder correction. In gaming and animation industries, action recognition can provide a natural human-computer interface and realistic motion synthesis.

An "action" is often considered as the smallest recognizable and meaningful motion unit. The task of human action recognition is to classify an input sequence of observations into one of the pre-trained actions. The common input includes RGB images, depth maps and skeletons. This paper focuses on skeleton data captured by a single Kinect sensor. As illustrated in Fig. 1, the skeleton data consists of 20 joints.

A typical action recognition process includes action representation and action classification [1], [2]. One way to represent an action either locally or globally is to use the spatio-temporal features including histogram of optical flow [3], 3D histogram of oriented gradients [4], motion history and energy images [5], [6]. Pose, inspired by the fact that human can easily recognize an action by looking at a few poses rather than the entire sequence, has also attracted attentions as another effective representation. Since not all poses in video sequences of an action are informative for the classification of that action, key poses are often considered when pose representation is adopted. The key poses of an action are expected to be representative for that action but at the same time discriminative from key poses of other actions. Existing methods following this approach often use K-means for key pose extraction. However, K-means algorithm may produce neutral poses which are common in all actions. This is due to the fact that K-means is an unsupervised learning method and thus the discriminative power of the poses is not considered.

In this paper, a method, referred to as ELC-KSVD, for extracting discriminative key poses from 3D skeleton data is proposed by extending the label consistent K-SVD (LC-KSVD) dictionary learning method. An action recognition framework as shown in Fig. 2 is developed upon the ELC-KSVD. First, a moving pose descriptor [7] is extracted from each skeleton frame of action samples. The proposed ELC-KSVD is then applied to learn discriminative key poses. The ELC-KSVD method aims to learn a common dictionary and multiple action-specific dictionaries. The atoms of common dictionary will be shared by most action classes whereas the atoms of action-specific dictionaries are specific to actions. Therefore, the common part is informative for representation but not discriminative for recognition. However, action-specific part is discriminative among classes. Consequently, the learned action-specific dictionaries can be considered as discriminative key poses and used to obtain sparse codes of each frame. Action-based features are constructed by max pooling of the frame-based sparse codes through temporal pyramid model [8]. Those features are used to train a SVM classifier which is finally used for action recognition.

The main contributions of this paper include the extended LC-KSVD for learning the common poses and action-specific poses and the action recognition method built upon the ELC-KSVD. The remainder of this paper is organized as follows. Section II briefly reviews the related works on key poses extraction and action recognition using sparse representation. Section III describes the proposed method including formulation and optimization of ELC-KSVD and the temporal pyramid

978-1-4799-5409-4/14/$31.00 ©2014 IEEE

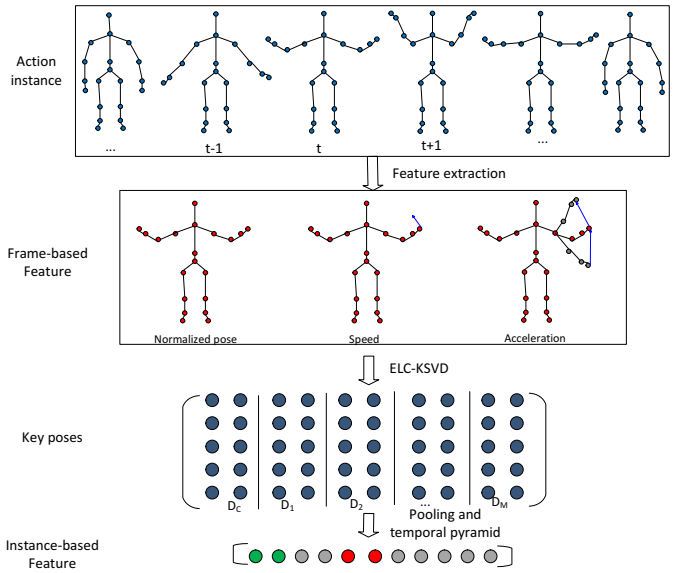

Fig. 2. The framework of the proposed method.

for constructing action-based descriptors. Experiments on the MSRC-12 gesture dataset and MSR-Action 3D dataset are presented in the Section IV. Section V concludes the work of this paper.

II. Related Work

This section provides a brief review on the recent works on key pose learning and sparse representation for action recognition.

For learning and extracting key poses, different methods including K-means and local maximal or minimal energy are often used, such as the works reported in [9], [10], [11]. Those methods can extract common poses shared by various action types and thus cause confusion for the classification phase. To overcome this problem, research on discriminative key poses extraction has been carried out. Methods of discriminative key poses extraction can be divided into three approaches. In the first approach, the discriminate key poses are extracted as the result of quantization. For example, Baysal et al. [12] used K-medians algorithm to extract candidate key poses. The final discriminative key poses were then determined based on the number of mapped frames in the same class with the key poses. Cheema et al. [13] adopted K-means to cluster poses of every action after representing poses by the contours of the silhouettes extracted from training video. Video frames were then mapped to the centers of clusters. The discriminative key poses were selected accordingly with the percentage of frames within the class. The second approach is to extract discriminative key poses according to centrality measure using page-rank. For instance, in Cao et al. [14], the discriminative key poses were selected using multistep page-rank through updating the probability of choosing the key poses as the final discriminative key poses. The third approach of extracting discriminative key poses is based on the classification results. Specifically, the key poses are those that have high contribution to classification. Weinland et al. [15] extracted a set of discriminative poses from all actions using forward selection and Liu et al. [16] adopted AdaBoost algorithm for discriminative key poses extraction. Of the three approaches, the last two often involve iterative process so they have relatively high computation. The first approach has computational advantage, involving two sequential steps: quantization and discriminative selection. Our method is similar to the first approach. But, unlike the first approach, it carries out the two steps simultaneously.

Sparse representation based on an overcomplete dictionary has been successfully applied to many computer vision problems, such as face recognition, image classification and object detection. A classical dictionary learning algorithm, K-SVD [17], was designed to minimize a Frobenius norm loss function and impose a sparsity constraint simultaneously. As K-SVD aims to minimize representation error and not classification error, discriminative K-SVD (D-KSVD) [18] and label consistent K-SVD (LC-KSVD) [19] were later developed to impose discriminative constraints on K-SVD to account for the classifier training cost and to construct one dictionary per class. Despite the fact that the class-specific dictionaries can represent class samples with minimum reconstruction errors, different classes usually share some common patterns which do not contribute to the classification. Therefore, DL-COPAR [20] was recently proposed to learn common patterns and class-specific patterns, which takes the discriminative class-based dictionary learning a step forward.

For action recognition, promising results have also been reported using sparse representation. A single overcomplete dictionary is learned by minimizing the representation error and constraining the sparsity using the classical model K-SVD in [21], [22], [23]. Meanwhile, multiply dictionaries are also learned for action recognition. Specifically, the class-specific dictionaries are independently learned. For instance, Guha et al. [24] learned three types dictionaries from the local motion pattern descriptor which include a single dictionary, a set of sub-dictionaries for every class and the concatenation of sub-dictionaries. Experiments showed that the concatenation of sub-dictionaries could achieve better results than the other two. Allfaro et al. [25] clustered action sequences of each action class into K clusters and learned K dictionaries in each action class.

Like D-KSVD [18] and LC-KSVD [19], discriminative dictionaries are also learned for action recognition. For example, Wang et al. [26] learned a dictionary for each class using the similarity-constraint and dictionary-incoherence. The similarity is computed based on the distance of test data to every dictionary. The dictionary-incoherence is measured by the inner-product of two different dictionaries. Luo et al. [8] proposed group sparsity and geometry-constraint dictionary learning (DL-GSGC) for action recognition from skeleton data. The group sparsity constrains the samples to use the atoms that belong to their class. The geometry-constraint forces features from same class to have similar coefficients as the coefficients of the pre-defined template features from every class. Although dictionary learning has been successfully applied in action recognition, common and class-specific dictionaries like DL-COPAR [20] have not been explored before. Analyzing the constitution of action, the separation of common and class-specific dictionaries is useful for classification. Since training samples often start from a neutral pose and end at a neutral pose, the neutral pose is shared by all action classes. Hence,

the common poses are not discriminative for classification. Based on this intuition, we extend the ELC-KSVD method to simultaneously learn a common dictionary and action-specific dictionaries for key pose extraction and then action recognition. Though DL-COPAR [20] was proposed for the same purpose, our experiments have shown the advantages of using ELC-KSVD over DL-COPAR in action recognition.

III. Proposed Method

The proposed method consists of three stages, frame-based feature extraction from skeleton data, the extended label consistent K-SVD (ELC-KSVD) dictionary learning, representation of actions and classification.

A. Frame-based Feature Representation

For each frame of skeletons, a moving pose descriptor [7] is extracted, which consists of normalized 3D joint locations, their velocities and accelerations. The velocity captures the moving direction of the joints which is essential to distinguish actions with similar poses but different order of poses. The acceleration captures the change in velocity over time.

Let $P(t) = [P_1, P_2, ..., P_N]$ represent N joints at time t, where $P_i = (p_x, p_y, p_z)$ is the 3D coordinates of joint i. The moving pose descriptor for frame t is denoted as $[\overline{P}(t), \beta_1 \delta\overline{P}(t), \beta_2 \delta^2\overline{P}(t)]$, where $\overline{P}(t)$, $\delta\overline{P}(t)$ and $\delta^2\overline{P}(t)$ are the normalized locations, velocity and acceleration of the joints respectively; β_1 and β_2 are the parameters to weight the importance of the velocity and acceleration components.

Pose normalization aims to eliminate the influence of subjects and camera position [7]. $\overline{P}(t)$ is obtained through selecting hip center (as shown in Fig. 1) as a reference of other joints and unifying the length of joint segments in different subjects. $\delta\overline{P}(t)$ can be calculated as the differences of same joints between the previous frame at $t-1$ and the next frame at $t+1$, the acceleration can be calculated using a temporal window of 5 frames around the current frame t

$$\begin{aligned} \delta\overline{P}(t) &= \overline{P}(t+1) - \overline{P}(t-1) \\ \delta^2\overline{P}(t) &= \delta\overline{P}(t+1) - \delta\overline{P}(t-1) \\ &= \overline{P}(t+2) + \overline{P}(t-2) - 2\overline{P}(t) \end{aligned} \tag{1}$$

For a Kinect skeleton, there are 20 joints and the moving pose is a descriptor of dimension 180. Notice that, since the calculation of the acceleration $\delta^2\overline{P}(t)$ requires the joints of frame $t-2$ and frame $t+2$, the first two frames and the last two frames of an action instance are ignored.

B. Extended Label Consistent K-SVD Dictionary Learning

1) Proposed ELC-KSVD: The Extended Label Consistent K-SVD (ELC-KSVD) is an extension of LC-KSVD. It also uses the label information of the input feature vectors. Unlike LC-KSVD, it aims to learn a common dictionary for all classes and a specific dictionary for each class. The key idea is to use label information to restrict the use of dictionary atoms.

Let Y be a set of d-dimensional N feature vectors of input skeleton frames in an action instance as $Y = [y_1, y_2, ..., y_N] \in R^{d*N}$. The sparse representation of Y through learning a dictionary D can be accomplished by solving the following problem:

$$\begin{aligned} \{D, X\} = \arg\min_{D,X} \parallel Y - DX \parallel_2^2, \\ \text{s.t.} \forall i, \parallel x_i \parallel_0 \leqslant T \end{aligned} \tag{2}$$

where $D = [d_1, d_2, ..., d_K] \in R^{d*K}$ is the learned dictionary which contains K atoms, $X = [x_1, x_2, ..., x_N] \in R^{K*N}$ is the sparse code of input Y and T is the sparsity constraint factor. The construction of D is the process of minimizing the reconstruction error $\parallel Y - DX \parallel_2^2$ in the constraint of sparsity. Based on the reconstruction error, a better representation of Y can be achieved at the cost of inferior classification.

To obtain a classification-oriented dictionary, an intuitive way is to learn M class-specific dictionaries $D_s (s \in [1, M])$ for all M classes. Furthermore, the class-specific dictionaries D_s from different categories usually share some common atoms which do not contribute to classification but are essential for reconstruction in dictionary learning. For example, different actions share neutral poses before and after motion. Hence, the common atoms should be separated by learning the commonality D_C for all actions to improve classification performance. Denote the overall dictionary as $D = [D_C, D_S] = [D_C, D_1, D_2, ..., D_M] \in R^{d*K}$ where $K = \sum_{s=1}^{M} K_s + K_C$, $D_s \in R^{d*K_s}$ denotes the specific dictionary of the s^{th} class and $D_C \in R^{d*K_C}$ stands for the common dictionary.

D_C and D_S are derived by introducing the common label information Q_C and class-specific label information Q_S. Q_C forces that frames of most actions are represented by the common dictionary D_C. Q_S enforces that action frames from s^{th} class use atoms from D_s. The objective function is then written as

$$\begin{aligned} \{D, X, A\} = \arg\min_{D,X,A} \parallel Y - DX \parallel_F^2 + \alpha \parallel Q - AX \parallel_F^2, \\ \text{s.t.} \forall i, \parallel x_i \parallel_0 \leqslant T, \quad T = 2 \end{aligned} \tag{3}$$

where $Q = [Q_C; Q_S] \in R^{K*N}$ is the conjunction of common label and discriminative label of Y. $Q_C = [q_1^C, q_2^C, ..., q_N^C] \in R^{K_C*N}$ where K_C is the number of atoms in common dictionary D_C, $q_i^C = [1, 1, ..., 1]^t \in R^{K_C}$ and non-zero values represent the samples are allowed to use the corresponding atoms. $Q_S = [q_1^1, q_2^2, ..., q_N^M] \in R^{K_s*N}$ represents label of all action classes of Y. The value $q_i^s = [0, 1, 1, 0..., 0]^t \in R^{K_s}$ is the class-specific label corresponding to y_i and non-zero values denote y_i uses this atom. Thus, Q can be written as

$$Q = \left[\begin{pmatrix} q_1^C \\ q_1^1 \end{pmatrix}, \begin{pmatrix} q_2^C \\ q_2^2 \end{pmatrix}, ..., \begin{pmatrix} q_N^C \\ q_N^M \end{pmatrix} \right]$$

The value A in Equation 3 represents the transformation matrix which forces the class-specific sparse code to be the most discriminative between different categories. The term $\parallel Q - AX \parallel_F^2$ represents the discriminative sparse error which forces X to approximate Q. On this constraint, frames of class s will be constructed by the atoms from D_C and D_s and not from $D_{\notin s}$. The atoms of D_s will form the key poses and atoms of D_C represent neutral or common poses and may account for reconstruction error. As pose of every frame may belong to a neutral pose or one of the key poses, the sparsity T is defined as 2 in order to force each frame is reconstructed by one common atom and a class-specific atom.

The following example will further show the constraint on D. Suppose there are 6 frames which belong to 3 classes, $K_C = 3$ and the number of class-specific dictionary atoms $K_1 = 2$, $K_2 = 1$, $K_3 = 1$. $Q \in R^{7*6}$ which is defined as

$$Q = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

The first three rows of Q are the common sparse codes and others are class-specific sparse codes. From this we can see that frames y_1 and y_2 use the first two atoms in the specific dictionary which belong to class 1. Frames y_3, y_4 and y_5 from the class 2 use the third atoms and y_6 from class 3 uses the last atom. On this constraint, frames from s^{th} class will use atoms of D_s and they are expected to have similar sparse codes.

2) Optimization Step: The efficient K-SVD algorithm [17] and the orthogonal matching pursuit (OMP) algorithm [27] are used to find the optimal solutions of D and X. Equation 3 can be rewritten as

$$\{D, X, A\} = \arg\min_{D,X,A} \parallel \begin{pmatrix} Y \\ \sqrt{\alpha}Q \end{pmatrix} - \begin{pmatrix} D \\ \sqrt{\alpha}A \end{pmatrix} X \parallel_F^2, \quad \text{s.t.} \forall i, \parallel x_i \parallel_0 \leqslant T, \quad T = 2 \tag{4}$$

Let $Y' = (Y, \sqrt{\alpha}Q)^t$, $D' = (D, \sqrt{\alpha}A)^t$ which is a normalized column-wise matrix. Thus, optimization of Equation 4 is equivalent of finding the solution of the following problem

$$\{D', X\} = \arg\min_{D',X} \parallel Y' - D'X \parallel_F^2, \quad \text{s.t.} \forall i, \parallel x_i \parallel_0 \leqslant T, \quad T = 2. \tag{5}$$

As this problem is exactly the classical K-SVD problem [17], this paper will follow the solutions of K-SVD problem. Given the initialized dictionary D_0 and the transformation matrix A_0, the sparse code X_0 is calculated using the OMP algorithm [27]. Then the dictionary is updated using single value decomposition (SVD) and sparse codes are updated using the OMP algorithm. To obtain fast convergence, the sparse codes are also updated in the dictionary updating stages. Denote d_k as an atom of dictionary and x_k which is the k^{th} row in X as the corresponding sparse code. The update of d_k and x_k is realized by solving the following problem

$$\{d_k, \widetilde{x}_k\} = \arg\min_{d_k, \widetilde{x}_k} \parallel \widetilde{E}_k - d_k \widetilde{x}_k \parallel_F^2 \tag{6}$$

where

$$E_k = Y - \sum_{j \neq k} d_j x_j \tag{7}$$

$\widetilde{x}_k$ and $\widetilde{E}_k$ represent the results of discarding the zero entries in x_k and E_k. Then SVD

$$U \Sigma V^t = SVD(\widetilde{E}_k) \tag{8}$$

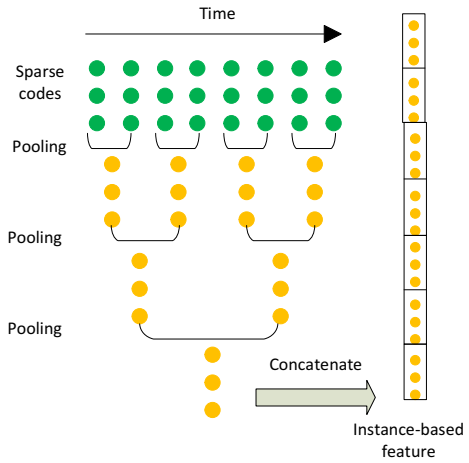

Fig. 3. Temporal pyramid for constructing an instance-based feature.

is used to find the solution of Equation 6 as,

$$d_k = U(:, 1) \quad \widetilde{x}_k = \Sigma(1, 1) V(:, 1) \tag{9}$$

C. Instance-based Feature Representation and Recognition

When the dictionary D is learned, the sparse code x of frame-based feature y of an action instance can be calculated by solving problem using OMP algorithms [27]

$$min_x \parallel y - Dx \parallel_2^2, \quad \parallel x \parallel_0 = 2 \tag{10}$$

Since D is learned using sparsity of 2, here $\parallel x \parallel_0$ also uses 2 .

As action instances contain different number of frames, the instance-based feature needs to be constructed in order to facilitate action classification. The instance-based feature is obtained based on the frame-based sparse code by taking into consideration the temporal dynamics. Here, three-level temporal pyramid is adopted to obtain the instance-based feature. The instance which contains T frames is divided into 2^{l-1} segments in the l^{th} level along the temporal information. Each segment in the l^{th} level contains $\frac{T}{2^{l-1}}$ frames. The sparse code of each segment is obtained using max pooling method [28]. Then sparse codes of all segments are concatenated to form an action- or instance-based feature vector. The process is illustrated in Fig. 3. The green parts are frame-based sparse codes in an instance. Each column in green color stands for a sparse code of a frame. Suppose the number of atoms in dictionary D is K, the dimension of an instance-based sparse code is $K \sum_{l=1}^{3} 2^{l-1} = 7K$.

Once features obtained for all training and test samples are obtained, a non-linear lib-SVM classifier [29] is trained for action recognition.

IV. Experiments

Two action datasets, MSRC-Kinect gesture dataset [30] and MSR-Action 3D dataset [2], were used to evaluate the proposed method. Both datasets were acquired using a Kinect sensor. Experiments show that key poses can be extracted from skeleton data using the proposed method. Furthermore, the

method was compared with other state-of-the-art methods on these two datasets to show its effectiveness.

A. Experimental Setup

In the proposed method, there are five parameters. In the frame-based feature extraction phase, the parameters β_1 and β_2 were set to 0.75 and 0.6 which are the same as those in Zanfir [7].

In the ELC-KSVD dictionary learning phase, the parameter α ranged from 0.001 to 0.5 which keeps the two terms in Equation 4 balanced. As the set of common dictionary is the common pose such as neutral pose or reconstruction error, the number of atoms in common dictionary was set from 20 to 70. When it is too large, the class-specific dictionaries may not be separated from the common dictionary. The class-specific dictionaries are to be the discriminative key poses, so the number of atoms in each class-specific dictionary was set from 3 to 5. In addition, dictionary D_0 and transformation matrix A_0 need to be initialized and label information Q also need to be given. D_0 was obtained using several iterations of K-SVD [17]. The common part of D_0 was learned from the skeletons of all action classes while the class-specific part was learned from skeletons of each class. Then the atoms of common and class-specific dictionaries were combined to obtain D_0. As we observed most instances in these two datasets often start and end with neutral poses, in the step of the K-SVD initialization, the common atoms were randomly selected from the beginning and ending frames of all training samples. The atoms of class-specific dictionaries were randomly selected from the middle frames of the instances of each class.

The label information Q was set based on the training samples. It is a constant in the dictionary learning process given the training samples. To initialize A_0, the spare code X of the training Y was calculated using the K-SVD [17] with the initialized D_0. The multivariate ridge regression model [31] was employed to obtain A_0 based on the values of X and Q.

$$A_0 = arg\ min_A \parallel Q - AX \parallel^2 + \lambda \parallel A \parallel^2 \qquad (11)$$

The solution of A_0 is shown as

$$A_0 = (XX^t + \lambda I)^{-1} XQ^t \qquad (12)$$

B. MSRC-12 Gesture Dataset

MSRC-12 gesture dataset [30] contains 594 video sequences and 719359 frames. It was collected from 30 subjects performing 12 gestures. Each sequence contains several instances of one gesture performed by one subject for several times. The gesture classes are divided into two groups including metaphoric gestures and iconic gestures. Table I lists the 12 gesture classes and the number of instances in each class.

To test the proposed method, single action instances were manually segmented from the 594 video sequences so that each instance contains one performance of a gesture. There are 6244 such instances. Here, the middle position between the end of previous performance and the start of the next performance along time was simply used to split the sequences into action instances. Therefore, each action instance often contains a large number of neutral or static poses at the beginning and ending. This is very different from the work reported in [32] which only kept the motion frames when the sequences were split into action instances.

TABLE I. GESTURE CLASSES IN MSRC-12 DATASET.

Metaphoric Gestures	No. of Instances
Lift outstretched arms	508
Push right	522
Wind it up	649
Bow	507
Had enough	508
Beat both	516
Iconic Gestures	**No. of Instances**
Duck	500
Goggles	508
Shoot	511
Throw	515
Change weapon	498
Kick	502

In the experiment, all 6244 instances were used for evaluating the proposed method. In order to compare with the state-of-the-art algorithms, cross subject test was conducted. Half of the subjects were randomly selected for training and the rest for testing. The experiment was repeated 20 times and the average results are reported below.

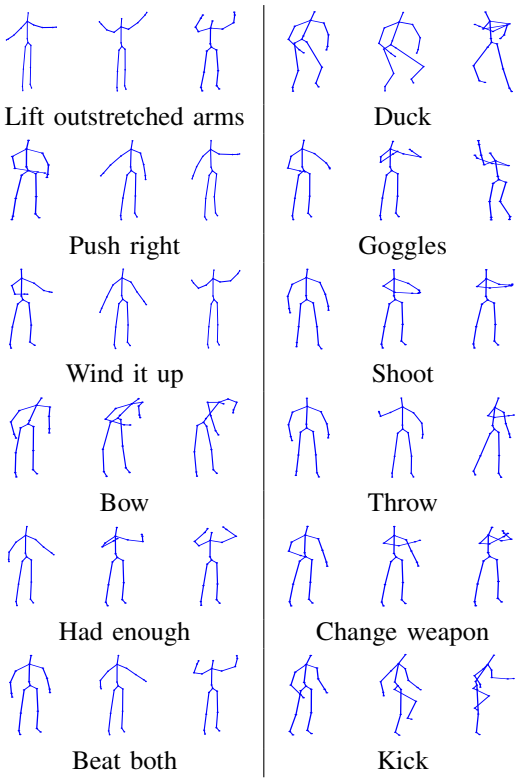

Fig. 4. The extracted key poses of 12 gestures in MSRC-12 using ELC-KSVD.

Fig. 4 shows the extracted key poses of 12 gestures using

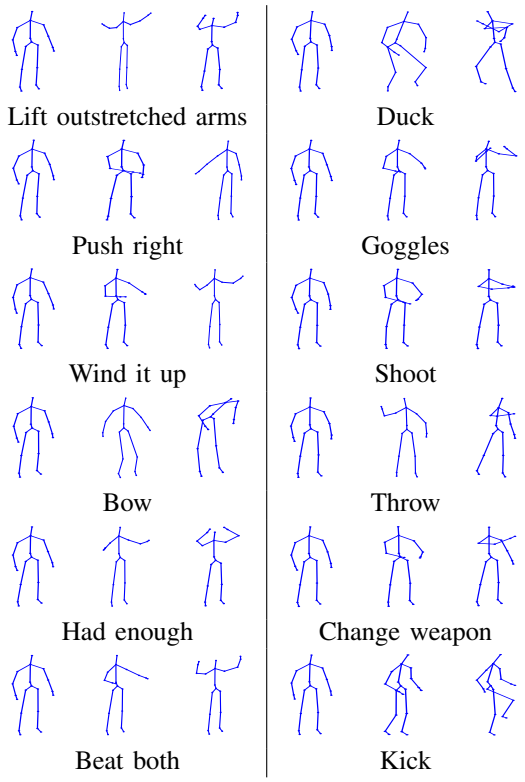

Fig. 5. The extracted key poses of 12 gestures in MSRC-12 using DL-COPAR [20].

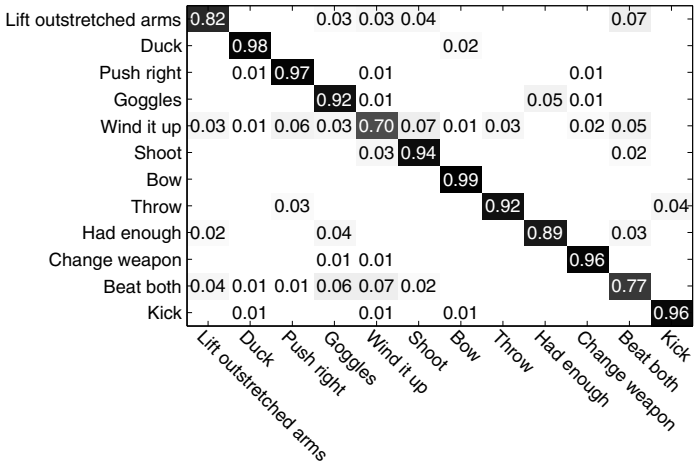

Fig. 6. Confusion matrix on the MSRC-12 dataset.

the proposed method. It can be seen that the three key poses are effective in all gestures except the third pose of gesture "Duck". This key pose probably reflect the noise in "Duck". Kong et al. [20] separated the commonality and particularity based on DL-COPAR. For comparison, DL-COPAR was used to learn the common and class-specific dictionaries under the same experimental setting. Fig. 5 shows the extracted key poses using DL-COPAR. Obviously, there is a neutral pose in the set of key poses for every action which appears in the first position of each set of key poses in Fig. 5. This happens much less when the proposed ELC-KSVD was used to learn the key poses. Notice that in our method there seems to be a neutral pose which is the first pose in the key poses for "Shoot","Throw" and "Beat".

TABLE II. RECOGNITION ACCURACY OF CROSS SUBJECT TEST IN MSRC-12.

Methods	Accuracy
Cov3DJ [32]	91.70%
DL-COPAR [20]	83.81%
ELC-KSVD	**90.22%**

Action recognition was conducted using the learned key poses. The average accuracy is 90.22% and confusion matrix is presented in Fig. 6. From the confusion matrix it can be seen that action "Lift outstretched arms", "Wind it up" and "Beat" are likely to be confused. This is because they are highly similar. Compared the result of the state-of-the-art work [32], the performance of our method is slightly worse as shown in Table II. This is mainly because that the neutral poses were manually removed in [32], while they are not removed in our experiments. Moreover, result of ELC-KSVD is better than DL-COPAR by approximately 7%.

C. MSR-Action 3D Dataset

MSR-Action3D dataset [30] contains 567 action instances with 23797 frames. It was collected from 10 subjects performing 20 actions for 2 or 3 times. The 20 actions are *High arm wave*, *Horizontal arm wave*, *Hammer*, *Hand catch*, *Forward punch*, *High throw*, *Draw x*, *Draw tick*, *Draw circle*, *Hand clap*, *Two hand wave*, *Side boxing*, *Bend*, *Forward kick*, *Side kick*, *Jogging*, *Tennis swing*, *Tennis serve*, *Golf swing*, *Pick up & throw*. As there are 23 instances whose skeleton data are zeros, only 544 instances were used in our experiments.

For a fair comparison, the experiments setting is same to the state-of-art algorithm. Half subjects (subject 1, 3, 5, 7, 9) were used for training and the rest for testing. Fig. 7 shows the extracted key poses using the proposed method. From the figure, most extracted key poses well capture the actions. As all actions in this dataset have neutral poses at the beginning and end of instances, the neutral poses should be separated with the action-specific poses. The results show only two actions "Forward punch" and "High throw" still include a neutral pose which appears in the first position of each set of the extracted key poses. The recognition accuracy using key poses is 88.70%. Compared to the state-of-art methods on this dataset shown in Table III, it is better than the results reported in [33] and slightly worse than [7] and [32]. This is probably because the dictionary sizes were too small. However, it outperforms the performance obtained by using DL-COPAR.

V. CONCLUSION

In this paper, we have extended the label consistent K-SVD for discriminative key pose extraction from 3D skeleton data. Discriminative key poses were obtained from the learned action-specific dictionaries. The proposed key pose extraction method was evaluated on two benchmark action datasets.

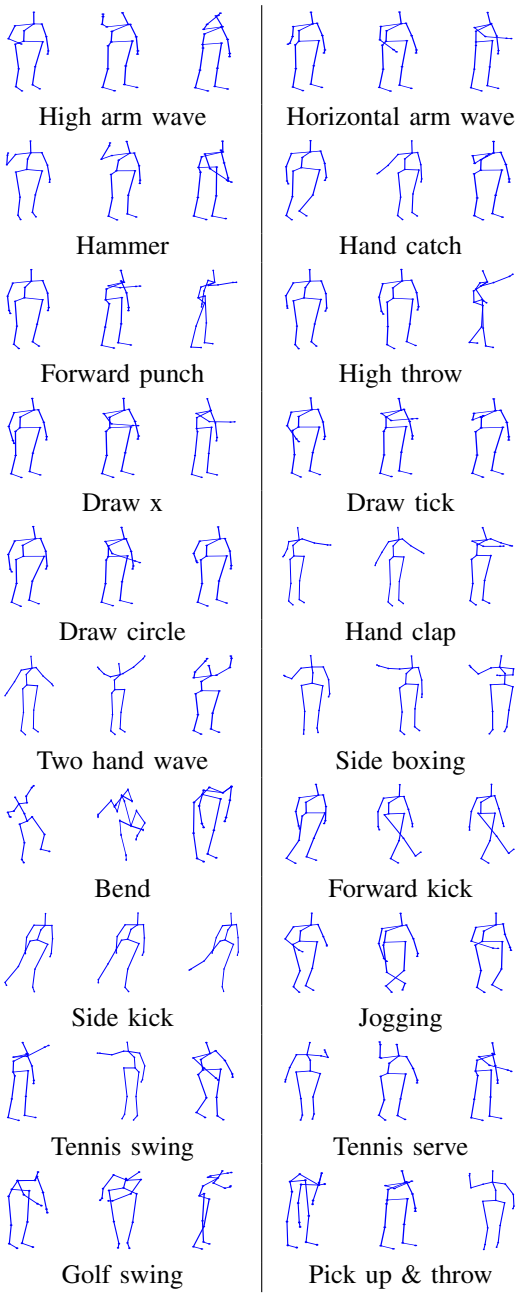

Fig. 7. The extracted key poses of 20 actions in MSR-Action3D using ELC-KSVD.

TABLE III. RECOGNITION ACCURACY OF CROSS SUBJECT TEST IN MSR-A3D.

Methods	Accuracy
Moving pose [7]	91.70%
Cov3DJ [32]	90.53%
Actionlet ensemble [33]	88.20%
DL-COPAR [20]	86.92%
ELC-KSVD	**88.70%**

Experimental results have shown that the proposed method is effective to extract key poses and the proposed ELC-KSVD has better performance than DL-COPAR in separating the common and class-specific patterns (dictionaries).

REFERENCES

[1] W. Q. Li, Z. Y. Zhang, and Z. C. Liu, "Expandable data-driven graphical modeling of human actions based on salient postures," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 11, pp. 1499–1510, 2008.

[2] ——, "Action recognition based on a bag of 3d points," in *IEEE International Workshop on Computer Vision and Pattern Recognition for Human Communicative Behavior Analysis*, 2010.

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

[4] A. Klaser and M. Marszalek, "A spatio-temporal descriptor based on 3D gradients," in *Proceedings of the 19th British Machine Vision Conference*, 2008.

[5] A. Bobick and J. Davis, "The recognition of human movement using temporal templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, pp. 257–267, 2001.

[6] D. Weinland, M. Ozuysal, and P. Fua, "Making action recognition robust to occlusions and viewpoint changes," in *Proceedings of the 11th European Conference on Computer Vision*, 2010.

[7] M. Zanfir, M. Leordeanu, and C. Sminchisescu, "The moving pose: An efficient 3D kinematics descriptor for low-latency action recognition and detection," in *Proceedings of 2013 IEEE International Conference on Computer Vision*, 2013.

[8] J. J. Luo, W. Wang, and H. R. Qi, "Group sparsity and geometry constrained dictionary learning for action recognition from depth maps," in *Proceedings of the 14th IEEE International Conference of Computer Vision*, 2013.

[9] W. J. Gong, A. D. Bagdanov, F. X. Roca, and J. Gonz'alez, "Automatic key pose selection for 3d human action recognition," in *Proceedings of the 6th International Conference on Articulated Motion and Deformable Objects*, 2010.

[10] W. J. Gong, J. Gonz'alez, and F. X. Roca, "Human action recognition based on estimated weak poses," *EURASIP Journal on Advances in Signal Processing*, vol. 1, pp. 1–17, 2012.

[11] A. A. Chaaraouia, P. Climent-Pereza, and F. Florez-Revuelta, "Silhouette-based human action recognition using sequences of key poses," *Pattern Recognition Letters*, vol. 34, no. 15, pp. 1799–1807, 2013.

[12] S. Baysal, M. C. Kurt, and P. Duygulu, "Recognizing human actions using key poses," in *Proceedings of International Conference on Pattern Recognition*, 2010.

[13] S. Cheema, A. Eweiwi, C. T., and C. Bauckhage, "Action recognition by learning discriminative key poses," in *ICCV Workshop on Performance Evaluation on Recognition of Human Actions and Pose Estimation Methods*, 2011.

[14] X. B. Cao, B. Ning, P. K. Yan, and X. L. Li, "Selecting key poses on manifold for pairwise action recognition," *IEEE Transactions on Industrial Informatics*, vol. 8, no. 1, pp. 168–177, 2012.

[15] D. Weinland and E. Boyer, "Action recognition using exemplar-based embedding," in *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, 2008.

[16] L. Liu, L. Shao, X. T. Zhen, and X. L. Li, "Learning discriminative key poses for action recognition," *IEEE Transactions on Cybernetics*, vol. 43, no. 6, pp. 1860–1870, 2013.

[17] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing over-complete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, 2006.

[18] Q. Zhang and B. Li, "Discriminative K-SVD for dictionary learning in face recognition," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[19] Z. Jiang, Z. Lin, and L. S. Davis, "Learning a discriminative dictionary for sparse coding via label consistent K-SVD," in *Proceddings of IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[20] S. Kong and D. H. Wang, "A dictionary learning approach for classification: separating the particularity and the commonality," in *Proceedings of 12th European Conference on Computer Vision*, 2012.

[21] C. H. Liu, Y. Yang, and Y. Chen, "Human action recognition using sparse representation," in *Proceedings of IEEE International Conference on Intelligent Computing and Intelligent Systems*, 2009.

[22] K. Guo, P. Ishwar, and J. Konrad, "Action recognition using sparse representation on covariance manifold of optical flow," in *Proceedings of IEEE International Conference on Advanced Video and Signal Based Surveillance*, 2010.

[23] S. Bomma, N. M. Robertson, and P. Favaro, "Sparse representation based action and gesture recognition," in *Proceeding of IEEE International Conference on Image Processing*, 2013.

[24] T. Guha and R. K. Ward, "Learning sparse representation for human action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 8, pp. 1576–1588, 2012.

[25] A. Alfaro, D. Mery, and A. Soto, "Human action recognition from inter-temporal dictionaries of key-sequences," in *Proceedings of the 6th Pacific-Rim Symposium on Image and Video Technology*, 2013.

[26] H. R. Wang, C. F. Yuan, W. M. Hu, and C. Y. Sun, "Supervised class-specific dictionary learning for sparse modeling in action recognition," *Pattern Recognition*, vol. 45, no. 11, pp. 3902–3911, 2012.

[27] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proceedings of Conference Record of the Twenty-Seventh Asilomar Conference on Signals, Systems and Computers*, 1993.

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

[29] C. Chih-Chung and L. Chih-Jen, "LIBSVM: a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 27, pp. 1–27, 2011.

[30] S. Fothergill, H. mentis, P. Kohli, and S. Nowozin, "Instructing people for training gestural interactive systems," in *Proceedings of the SIGCHI Conference on Human Factors in Computing systems*, 2012.

[31] G. Golub, P. Hansen, and D. O'leary, "Tikhonov regularization and total least squares," *SIAM Journal on Matrix Analysis and Applications*, vol. 21, pp. 185–194, 1999.

[32] M. E. Hussein, M. Torki, M. A. Gowyyed, and M. El-Saban, "Human action recognition using a temporal hierarchy of covariance descriptor on 3D joint locations," in *Proceedings of the 23th International Joint Conference on Artificial Intelligence*, 2013.

[33] J. Wang, Z. C. Liu, Y. Wu, and J. S. Yuan, "Learning actionlet ensemble for 3D human action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. P, no. 99, pp. 1–14, 2013.