Human Action Recognition based on MSVM and Depth Images

Ahmed Taha[1], Hala H. Zayed [1], M. E. Khalifa[2] and El-Sayed El-Horbaty[3]

**[1] Computer Science Dept., Faculty of Computers & Informatics, Benha University,
Benha, Egypt.**

**[2] Basic Science Dept., Faculty of Computer & Information Sciences, Ain Shams University,
Cairo, Egypt.**

**[3] Computer Science Dept., Faculty of Computer & Information Sciences, Ain Shams University,
Cairo, Egypt.**

Abstract

Human behavior Analysis, using visual information in a given image or sequence of images, has been an active area of research in computer vision community. The image captured by conventional camera does not provide the suitable information to perform comprehensive analysis. However, depth sensors have recently made a new type of data available. Most of the existing work focuses on body part detection and pose estimation. A growing research area addresses the recognition of human actions based on depth images. In this paper, an efficient method for human action recognition is proposed. Our research makes the following contributions: the proposed method makes an efficient representation of human actions by constructing a feature vector based on the human's skeletal information extracted from depth images. Then, introducing these feature vectors to Multi-class Support Vector Machine (MSVM) to perform the action classification task. The proposed representation of the human action ensures it is invariant to the scale of the subjects/objects and the orientation to the camera, while it maintains the correlation among different body parts. A number of experiments have been performed in order to evaluate the proposed algorithm. The results revealed that the proposed algorithm is efficient and leads to an improved action recognition process. Moreover, it is suitable for implementation in a real-time behavior analysis.

Keywords: *Behavior Analysis, Video Surveillance, Action Recognition, Depth Images, Multi-class Support Vector Machine.*

1. Introduction

Human action analysis has been widely studied for many applications including visual surveillance, video retrieval, human computer interaction, and sports training [1-8]. Among all kinds of applications, intelligent surveillance systems make a great impact on daily lives by supporting healthcare and security service [9]. Behavior analysis is the advanced stage in intelligence surveillance. As the amount of video data collected daily by surveillance cameras increases, the need for automatic systems to detect and recognize different activities performed by people and objects also increases.

Recognizing human actions from video stream is not new. It has been around since the early days of computer vision [10]. Due to the large diversity of human body, size, appearance, posture, motion, clothing, view angle, camera motion, and illumination changes, besides the complexity of human actions, the task of automatically recognizing human actions is very challenging. Different persons will perform the same action in a different manner, and even the same person will perform it differently at different times [10].

Extensive research has been reported on human action recognition using features extracted from 2D intensity images [1-6]. In addition, Support Vector Machines (SVMs) and Hidden Markov Models (HMMs) have been widely applied to classify actions [11]. However, the 2D intensity image captured by conventional camera does not provide the suitable information to perform comprehensive analysis of human actions. In addition, it is sensitive to lighting changes and the process of detecting interest points depends heavily on the object texture rather than the object geometry [10]. Moreover, intensity images face many challenges to robustly perform computer vision tasks such as background subtraction and object segmentation.

Depth images overcome some of the limitations of intensity images. With the development of low-cost depth cameras, the computer vision field has experienced a new opportunity of applying a practical solution for building a diversity of systems in different fields. A depth camera provides depth information as different means to color images captured by the traditional optical cameras. Depth information gives extra robustness to color, as it is invariant to lighting and texture changes [12]. The depth camera provides information about the 3-D structure of the scene as well as the 3-D motion of the subjects/objects in the scene. Therefore, ambiguity of the conventional camera, i.e., projection of the 3-D physical world onto the 2-D image plane, could be avoided [12].

In this paper, a method for human action recognition is proposed. The proposed method presents an efficient representation of human actions by constructing a reduced feature vector based on the human's skeletal information extracted from depth images. This representation of the human action is invariant to the scale of the subjects/objects and the orientation to the camera, while it maintains the correlation among different body parts. In addition, the low dimensionality of the constructed feature vector makes this representation superior to its counterparts. The constructed feature vectors are then passed to Multi-class Support Vector Machine (MSVM) to perform the action classification task.

The remainder of this paper is structured as follows: section 2 gives an overview about depth images and their advantages over the intensity images. In section 3, we briefly review some related work in human action recognition. Section 4 then presents the proposed method. The performance analysis of the proposed method is empirically evaluated in Section 5. Finally, we conclude in Section 6.

2. Depth Image

A depth image (or depth map) is an image that contains information relating to the distance of the surfaces of scene objects from a viewpoint [13]. Pixels in a depth image indicate calibrated depth in the scene, rather than a measure of intensity or color. Depth images are captured by what is called a depth sensor or a depth camera. Depth cameras offer several advantages over traditional intensity sensors including [14]: working in low light levels, giving a calibrated scale estimate, being color and texture invariant, and resolving silhouette ambiguities in pose. They also greatly simplify many computer vision tasks such as background subtraction and object segmentation.

Depth imaging technology has seen a remarkable development in the last years especially after it reaches a consumer price point with the release of Microsoft Kinect. Recently, Microsoft Kinect (Microsoft Corporation) or the ASUS Xtion Pro (ASUSTeK Computer Inc.) has shown great reliability on capturing depth images. Their popularity comes from their low cost, high sample rate and capability of combining visual and depth information. Three main sensing technologies are applied in computer vision research to obtain depth images [10]: stereo cameras, Time-of-Flight (ToF) cameras, and structured light. The Kinect sensor uses the structured light to construct the depth map. The technology behind the Kinect sensor was originally developed by the PrimeSense Company, which released their version of an SDK to be used with the Kinect as part of the OpenNI (Open-source SDK for 3D sensors). Recently, ASUS Company also produces a sensor with the same capabilities as the Kinect sensor that works with the OpenNI SDK. Although these depth sensors were initially designed for gaming purposes, many other applications are extensively employing these technologies in both research and commercial fields [15].

Kinect consists of an RGB camera and a depth sensor. The depth sensor provides images at 640 × 480 pixels and 30 frames per second [16]. The Kinect sensor has a practical ranging limit of 0.8–4 meters, with a resolution of about a few centimeters. The Kinect depth sensor consists of an infrared projector and infrared CMOS sensor. An irregular pattern of dots (structured light) is projected onto the scene, and the depth measurement is based on triangulation [10]. By this mechanism, these image sensors capture both a depth image (D) and the regular color image (RGB). The resulting RGB-D data can be used to generate a skeleton model of objects in the scene. This characteristic data can be used later in order to learn and classify human poses, actions or even activities. Human skeleton extraction came originally with the need to estimate human poses. Later, human skeleton joints became one of the major analyzing characteristics in action recognition research [17, 18]. Therefore, one direct advantage of Kinect is its real-time availability of derived 3D skeleton joints for objects detected.

In order to obtain the skeletal information from depth maps, there are two major methods to extract the structured set of joints and their connections [15, 19]. One is Microsoft SDK by Microsoft (http://www.microsoft.com/en-us/kinectforwindows/); the other is OpenNI by PrimeSense (http://www.primesense.com/open-ni/). These two methods provide different kinds of skeleton models. The Microsoft Kinect SDK provides a skeleton model with 20 joints, whereas the OpenNI/NITE skeleton tracks a set of 15 joints. It should be mentioned that all 15 points of the OpenNI skeleton are a subset of 20 points of the Microsoft Kinect SDK skeleton. Figure 1 depicts the skeleton joints calibrated by both Microsoft SDK and OpenNI. All the solid and unsolid circles shown in the Figure show skeleton joints obtained by Microsoft SDK while the subset of solid circles only shows skeleton joints obtained by OpenNI. Both methods follow human pose detection, skeleton calibration, and then skeleton tracking.

Despite the superiority achieved by this type of cameras on traditional RGB cameras in different computer vision tasks, the applicability of a D-RBG camera is restricted by its limited range of depth in the range of 0.8 to 4 meters [18]. Once the depth exceeds that range, measurement errors will increase drastically. To overcome this problem, a RGB-D cameras network is constructed by deployment of several RGB-D cameras at various locations to extend the range of coverage. In [21], Han et al. present a reliable automatic localization technique in the RGB-D cameras network without the need for additional instrument or human intervention. This technique is employed to derive the relative location and orientation of a detected object taking into account precise localization of the camera network: relative location and orientation of neighboring cameras.

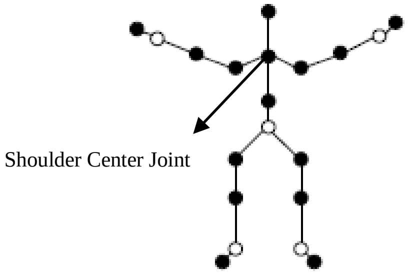


Fig. 1 [19] Skeleton joints detected by Microsoft SDK and OpenNI 2.1 Subheadings

3. Related Work

Most previous research in action recognition was based on color or greyscale intensity images. These images are obtained from traditional RGB cameras, where the value of each pixel represents the intensity of incoming light. It contains rich texture and color information, which is very useful for image processing, however it is very sensitive to illumination changes.

Recently, there have been vision technologies that can capture distance information from the real world, which cannot be obtained directly from an intensity image. These images are obtained from depth cameras, where the value of each pixel represents the calibrated distance between camera and scene [12]. An advantage of using these sensors is that they give depth at every pixel so the shape of the object can be measured. When using depth images, computer vision tasks like background subtraction and contour detection become easier. Actually, there are many attractive progresses and improves have been done with the use of depth information.

In this section, we review only the closely related work. Raptis et al. [19] propose an approach for human action recognition based on skeletal information by generating a spherical coordinate system in each frame. Their approach applies Principal Component Analysis (PCA) on the torso frame to get two orthonormal bases, which are aligned with the longer dimension of the torso (top-down) and with the line connects two shoulders (left-right). Two types of joints are considered to describe the position in the sphere coordinate system: first degree joints that are adjacent to the torso and second degree joints that are the extremities of the body except the head. The limbs points represented within this coordinate system are shown to be invariant to the orientation of the human body to the camera. However, the PCA calculation of the torso frame in each frame is computationally expensive, and it is not reliable when the line of the shoulders is orthogonal to the view of camera.

Zhu et al. [18] present a modification for the work presented by Raptis et al. [17]. They introduce a human action recognition solution, where the features of four limbs during actions are descripted as their sequences of angular representation. They believe that using only the second degree joints (hands, feet) provide more discriminate power than using the first degree joints (elbow, knees) with respect to different types of actions. Moreover, the first degree joints may have possibilities to introduce noise and make the recognition of an action incorrectly. Their angular representation of the limbs is described in the world coordinate system directly, as the orientation of each user during each action is calibrated to minimize the inconsistency. A modified Dynamic Time Warping (DTW) is then applied as a template matching solution to do the action classification task

In [15], a genetic based evolutionary algorithm is proposed to determine the optimal subset of skeleton joints, taking into account the topological structure of the skeleton, in order to improve human action recognition with RGB-D devices. The basic idea of their approach is to consider a binary vector where each gene represents the further consideration or not of a specific feature. Two main models are presented to implement this: the filter model, and the wrapper model. However, the approach suffers from two main difficulties: the high computational cost of the wrapper approach and early convergence. A wrapper-based evolutionary approach requires the calculation of the fitness of an important number of solutions (individuals) until the final solution is obtained. As a single fitness calculation involves a complete training and recognition process, the whole evolution could take considerable time. Moreover, early convergence happens when the evolutionary search is stacked in a local minimum and cannot achieve a good solution.

In [20], a method for human actions recognition from sequences of depth maps is proposed. The depth maps are projected onto three orthogonal planes and the whole sequence is accumulated generating a Depth Motion Map (DMM) similar to the Motion History Images (MHI). Then for each DMM, Histograms of Oriented Gradients (HOG) are obtained. The concatenation of the three HOG serves as input feature to a linear SVM classifier.

Yang et al. [21] propose a method to recognize human actions by extracting three features for each joint which are based on pair-wise differences of joint positions. The three features are differences between joints in the current frame; differences between joints in the current frame and the preceding frame; and differences between joints in the current frame and in the initial frame of the sequence, which approximates the neutral posture. As it can be noticed, the number of these differences results in a high dimensional feature vector. Hence, PCA is used to reduce redundancy and noise in the feature, and to obtain a compact EigenJoints representation for each frame. Finally, a naive-Bayes nearest-neighbor classifier is used for multiclass action classification.

Ong et al. [22] propose an unsupervised human activity detection not recognition with features extracted from skeleton data obtained from RGBD sensor. Their activity detection technique does not provide the activity label however; it attempts to distinguish one activity from another. The detection technique uses a suitable set of features with K-means clustering to distinguish different activities from a pool of unlabeled observations.

4. Proposed Method

In this section, the proposed method for human action recognition is presented. The basic idea of the proposed method depends on utilizing human's skeletal information extracted from depth images. This skeletal information is available by using depth sensors (e.g., Microsoft Kinect), where a skeleton data is generated for each human body recognized in the RGB-D data stream. First, a skeleton based action representation method is proposed. This representation of the human skeleton is invariant to the scale of the subjects/objects and the orientation to the camera, while it maintains the correlation among different body parts. Second, MSVM is used to perform action classification.

Figure 2 shows a block diagram of the proposed method while Figure 3 shows its pseudo code. Initially, the proposed method starts with identifying the skeleton joints coordinates for each detected object in the video sequence. Actually, the Kinect camera tracks 20 body joints for each object in the scene. The position of the skeleton joints are provided as Cartesian coordinates (X, Y, Z) with respect to a coordinate system centered at the Kinect. The positive Y axis points up, the positive Z axis points where the Kinect is pointing, and the positive X axis is to the left as shown in Figure 4.

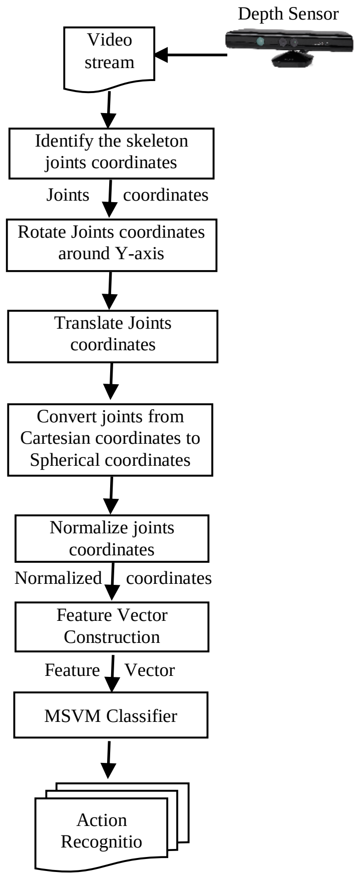

Depth Sensor
Video stream
Identify the skeleton joints coordinates
Joints coordinates
Rotate Joints coordinates around Y-axis
Translate Joints coordinates
Convert joints from Cartesian coordinates to Spherical coordinates
Normalize joints coordinates
Normalized coordinates
Feature Vector Construction
Feature Vector
MSVM Classifier
Action Recognitio

Fig. 2 The block diagram of the proposed method

Proposed Algorithm:

Inputs: video stream captured by depth sensor

Output: class Label for detected human actions
Steps:

```
1.  For each skeleton in the video sequence do:
        2.  Identify all joints coordinates
        3.  Determine the angle $\alpha$ between the line connecting
            both shoulders of the skeleton and the positive
            direction of X-axis of Kinect coordinates system
        4.  For each joint in the skeleton do:
                5.  Rotate the joint coordinate around Y-axis in a
                    counterclockwise direction with an angle $\alpha$
            End for
        6.  Set the origin coordinate of the skeleton to be
            shoulder center joint coordinate
        7.  For each joint in the skeleton do:
            8.  Translate the joint coordinate to the new system
                coordinate
            9.  Convert the joint Cartesian coordinate to its
                corresponding spherical coordinate
            10. Normalize the joint coordinate
            End for
```

```
11.  Construct feature vector
12.  Identify the class label by passing the feature vector to MSVM
```

Fig. 3 The pseudo code of the proposed method

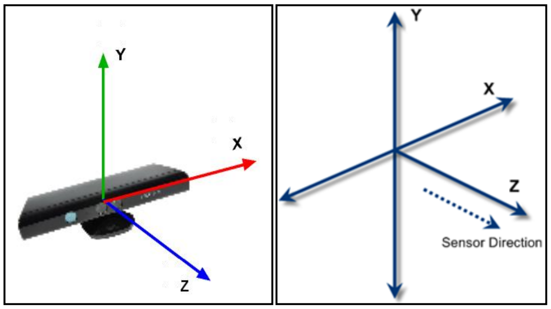


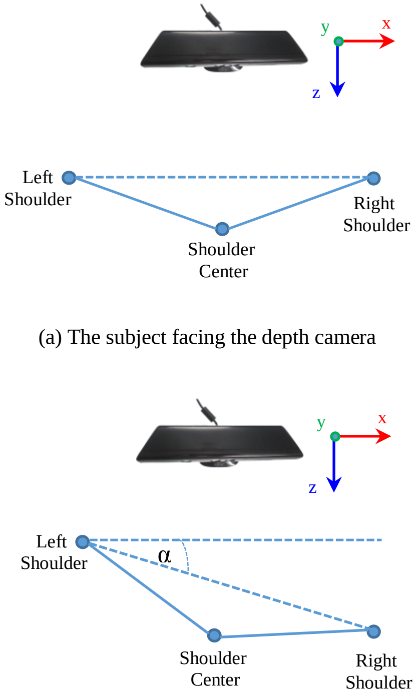
Fig. 4 Kinect Cartesian coordinate system

Ideally, a subject should be straight in front of Kinect camera (Figure 5.a) but this is not always the case. The subject can be at any angle from Kinect (Figure 5.b) and at any distance. To overcome this issue, the proposed method rotates all the skeleton points around Y-axis in a counterclockwise direction with an angle α in order to make the subject straight in front of depth camera. Hence, rotation invariance is achieved. This angle is defined as the angle between the line connecting both shoulders and the positive direction of X-axis of Kinect coordinates system (Figure 5.b). First, the angle α is estimated using the coordinates of two joints: shoulder left (x_L, y_L, z_L) and shoulder right(x_R, y_R, z_R) through the following equation:

$$\propto = \tan^{-1}\left(\frac{z_R - z_L}{x_R - x_L}\right)$$

Then a counterclockwise rotation about Y-axis is applied to all skeleton joints with an angle α. For each skeleton joint i with coordinates (x_i, y_i, z_i), the rotated coordinates (x_i', y_i', z_i') are calculated with the following transformation:

$$\begin{bmatrix} x_i' \\ y_i' \\ z_i' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\propto & 0 & \sin\propto & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\propto & 0 & \cos\propto & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix}$$



(a) The subject facing the depth camera

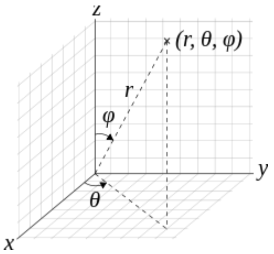
(b) The subject at an angle with respect to the depth camera

Fig. 5 Rotation of the skeleton with respect to the Kinect

Moreover, varying the object distance from Kinect makes the action recognition more sophisticated. Therefore, it is necessary to shift the origin of the coordinates from Kinect to a point in the object body to remove dependence on camera position. This means joints coordinates should be translated to another coordinate system where its origin is a point in the human body rather than the Kinect camera. By this way, the distance factor between the object and Kinect is neutralized. This permits the coordinates to be expressed invariantly to translation and rotation of the body with respect to the camera reference system. In our proposed method, we use the shoulder center joint as the origin of the new system (see Figure 1). Assume that shoulder center joint coordinates are (x, y, z). Hence for each skeleton joint i with coordinates (x_i, y_i, z_i), the translated coordinates (x_i', y_i', z_i') are calculated with the following equation:

$$(x_i', y_i', z_i') = (x_i - x, y_i - y, z_i - z)$$

Moreover, the individual variations of people in terms of posture, height and dimensions have a huge impact on the performance of the action recognition system. This is because X, Y and Z coordinates of joints of every object doing the same action might be different. Therefore, it is necessary to normalize the data to increase accuracy of action recognition. To simplify the normalization process, the joints coordinates are converted from Cartesian coordinate system to spherical coordinate system. The spherical coordinate system is a three dimensional space system with three components: the distance of the point from the origin (radial distance *r*), the polar angle (φ), and the azimuth angle (θ) as shown in Figure 6. When normalizing a point in Cartesian coordinates, all the components X, Y and Z are changed. However when normalizing a point in the spherical coordinates, only radial distance *r* will equal to one while both polar angle (φ) and azimuth angle (θ) will remain constant.



$$r = \sqrt{x^2 + y^2, z^2}\ , \quad \theta = \cos^{-1}\left(\frac{z}{r}\right), \quad \varphi = \tan^{-1}\left(\frac{y}{x}\right)$$

Fig. 6 Spherical coordinates (r, θ, φ): radial distance r, azimuthal angle θ, and polar angle φ

In fact, the representation method of actions should be expressive enough to describe a variety of actions yet sufficiently discriminative in distinguishing different actions. Most of the existing work that employs depth images for human action recognition use all the available joints obtained by the depth sensors devices [23]. However, actually not all joints contribute equally in defining the action. Therefore, it is important to determine which joints have a greater value to the success of the recognition process and which ones can be dropped. These latter joints not only have no effect in the recognition process but also reduce the recognition rate because they introduce confusion or noise.

Feature vectors provide a set of characteristics that represent the action to be recognized. However, it may include irrelevant or redundant information which could complicate the classification. Reducing the feature vector size has an important impact on the processing time since the recognition is performed faster. Concerning the skeletal data obtained with depth sensor devices, it can be seen that some joints are more important than others if action recognition is targeted. Several joints in the torso (the skeleton part identified by a dashed line in Figure 7) do not show an independent motion along with the whole body. Hence, in our proposed method, seven joints coordinates of the human skeleton are discarded from the feature vector. These joints are shown as solid circles in Figure 7: shoulder right, shoulder center, shoulder left, spine, hip center, hip right, and hip left (from left-to-right and from top-to-bottom respectively). This dimensionality reduction of the feature vector improves the classification performance. Since the joints coordinates are normalized, radial distance *r* can be ignored in our feature vector. Thus, the feature vector will consist of 13 pairs of (φ , θ) for each detected object in the scene. This means it has only 26 components which is a reduced feature vector than what is reported in the state-of-the-art methods [15, 20, 21]. A low-dimensional representation means less computational effort.

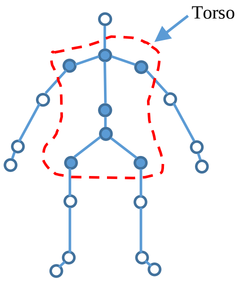


Fig. 7 Torso skeleton joints discarded from the feature vector

Once a feature vector is constructed, a classification step is needed to recognize different actions. The feature vector of the unknown action is used as input to the classifier whose objective is to accurately identify which action class is best matched against the input. In our proposed method, a Multi-class Support Vector Machine (MSVM) [24-27] is employed to perform action classification. The MSVM used is based on One-Against-All (OAA) classification approach [25] where there is one binary SVM for each class to separate members of that class from members of other classes. A data point would be classified under a certain class if and only if that class's SVM accepted it and all other classes' SVMs rejected it. A training step is needed to summarize the similarity within (and dissimilarity in-between) the training samples of different action classes. With action models learned, a new action instance can be recognized as one of the learned classes. This choice of MSVM is mainly justified by the fact that SVM has many advantages over other counterpart methods [24, 25, 27]: 1) providing better prediction on unseen test data, 2) providing a unique optimal solution for a training problem, and 3) having fewer parameters compared with other methods. In addition, it has been successfully applied to a wide range of pattern recognition and classification problems.

The SVM approach aims at finding the optimal separating hyperplane between classes by concentrating on the training cases that are placed at the edge of the class descriptors [25]. These training cases are called support vectors. Training cases other than support vectors are discarded. This way, not only is an optimal hyperplane fitted, but also less training samples are effectively used; hence high classification accuracy is achieved with small training sets. It has proved to have high generalization performance both theoretically and empirically [24, 26]. Obviously, using the appropriate classifier is important for successful recognition.

5. Experimental Results

In order to evaluate the performance of the proposed method, we use the Microsoft Action3D dataset [28]. It is a benchmark dataset used widely to evaluate the performance of RGBD-based action recognition methods [15, 20, 21]. The Microsoft Action3D dataset is a public dataset providing sequences of depth maps captured by a RGBD camera. The actions in this dataset catch a wide range of motions related to arms, legs, torso, and their combinations. It includes 20 different actions: 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, and pick up & throw [28, 29]. These actions are performed by 10 subjects facing to the camera during acting. Each action was performed two or three times by each subject. There are 567 depth map sequences in total. The depth maps are captured with a resolution of 320×240. The dataset uses the 20-joint model. For each skeleton joint, the horizontal and vertical positions are stored in screen coordinates, and depth value is stored in world coordinate.

In order to fairly evaluate the proposed method, we follow the same experimental settings used by most of the state-of-the-art methods [15, 20, 21]. As stated in [28], the dataset actions are divided into three subsets, each having 8 actions in order to reduce the computational cost of the tests. Table 1 shows the three subsets used in the experiments. Note that AS1 and AS2 group actions with similar movement, while AS3 groups complex actions together. We evaluate the proposed method using 2-fold cross validation (holdout method) in which each subset is divided randomly into two equal-size groups: one for training and the other for testing. The MSVM is trained using the training group while the other group is used for testing the method. After that, the two groups are used interchangeably and the method is evaluated again. The experiments were implemented on a 2.5GHz Intel Core i7 PC with 4GB memory, running under Windows 8 Enterprise. The algorithm is coded using MATLAB 8.1.0.604 (R2013a).

Table 1: Action subsets and tests used in the experiments [13,18, 19]

Action Set 1 (AS1)		***Action Set 2 (AS2)***		***Action Set 3 (AS3)***	
label	Action name	label	Action name	label	Action name
a02	Horizontal Wave	a01	High Wave	a06	High Throw
a03	Hammer	a04	Hand Catch	a14	Forward Kick
a05	Forward Punch	a07	Draw X	a15	Side Kick
a06	High Throw	a08	Draw Tick	a16	Jogging
a10	Hand Clap	a09	Draw Circle	a17	Tennis Swing
a13	Bend	a11	Hands Wave	a18	Tennis Serve
a18	Tennis Serve	a12	Forward Kick	a19	Golf Swing
a20	Pickup Throw	a14	Side Boxing	a20	Pickup Throw

Figure 8 shows the confusion matrices for the proposed recognition method for each action subset. Each row represents the instances in an actual class and each column denotes the recognition results. For example in the first row of Figure 8.a, 79.6% of the “horizontal wave” samples are classified correctly while 19.5% of the samples are misclassified as “hand clap” action and 0.9% are misclassified as “tennis serve” action. As, it can be seen from the figure, the results prove the efficiency of the proposed method in recognizing different actions. Although AS3 includes more complex actions than the other two subsets, the results for AS3 is better than those for AS1 and AS2. This is due to the great similarity between the actions in each subset of AS1 and AS2. For example in AS1, both “horizontal wave” action and “hand clap” action have similar movements. Also, “Pick-up and throw” is a complex action composed of two actions: “bend” and “high throw”. This complicates the recognition process because both actions are also included in AS1 subset. Moreover, the AS2 actions are very similar where all of them are performed by the arms except “forward kick” action.

	a02	a03	a05	a06	a10	a13	a18	a20
a02	79.6				19.5		0.9	
a03		100						
a05			96.2		3.8			
a06	2.2			92.5			5.3	
a10	15.2		0.2		84.6			
a13						100		
a18							100	
a20						2.7		97.3

(a) AS1 subset

	a01	a04	a07	a08	a09	a11	a12	a14
a01	76.1			1.6	9.1	13.2		
a04	10.2	84.6		5.2				
a07			99.4		0.6			
a08			1.3	98.7				
a09			2.7		97.3			
a11	0.2					99.8		
a12							100	
a14								100

(b) AS2 subset

	a06	a14	a15	a16	a17	a18	a19	a20
a06	85.4					5		9.6
a14		96.5	3.4	0.1				
a15		1.2	98.8					
a16				100				
a17					99.1	0.9		
a18			0.2		1.5	98.3		
a19							100	
a20								100

(c) AS3 subset

Fig. 8 Confusion matrix of the proposed method for each subset

In all previous experiments, the time taken to process each frame in each video sequence is calculated. Then, the average frame processing time (in milliseconds) is calculated for each action subset. Table 2 shows the rates that the proposed method are able to reach. The results reveal that the reduction in the size of the feature vector has an important effect in the computational cost of the recognition process. The reported results ensure that the proposed algorithm will be feasible for real time tracking applications.

Table 2: Average frame processing time (ms) of the proposed method

Dataset	***No of video sequences***	***Average Frame Processing Time (in milliseconds)***
AS1	220	1.7
AS2	228	2.1
AS3	222	1.9

In the evaluation of the proposed method, Table 3 shows the results of comparing the proposed method with the state-of-the-art methods [15, 20, 21]. The recognition accuracy rate of the proposed method can reach as high as 95.2%, which is superior to the other baselines. These results further verify the performance and the feasibility of the proposed method. The method presented in [15] suffers from both time and memory complexity incurred by the use of Dynamic Time Warping (DTW) as the recognition algorithm. Also, the computational cost of the DMM-HOG Descriptor presented in [20] is high. Each descriptor has a dimension of 7360, which forms a limitation to the use of the algorithm. In contrast, the feature vector of the proposed method is very low. It consists of only 26 values. In [21], Yang and Tian employ Naïve Bayes Nearest Neighbor (NBNN) as a classifier to recognize multiple action categories. Although NBNN is faster than k-nearest neighbor or support vector machines, its recognition accuracy is usually low. The results shown in Table 3 proves the drawback where the recognition accuracy of [21] is the lowest accuracy achieved comparing to the other methods.

Table 2: Recognition accuracies (%) of the proposed method compared to the state-of-the-art methods on MSR Action3D dataset.

Method	***Dataset***			***Average***
	AS1	***AS2***	***AS3***	
Yang and Tian (2012) [21]	74.5	76.1	96.4	82.3
Yang et al (2012) [20]	96.2	84.1	94.6	91.6
Chaaraoui et al (2014) [15]	91.5	91.8	97.1	93.5
The proposed method	**93.8**	**94.5**	**97.3**	**95.2**

6. Conclusions and Future Work

The problem of analyzing behaviors in video has been the focus of several researchers' efforts and several systems have been proposed in the literature. Recently released depth cameras provide effective estimation of 3D positions of skeletal joints in temporal sequences of depth maps.

This paper proposes a method for recognizing human actions. The proposed method presents a low dimensional representation of human actions by constructing a feature vector based on the human's skeletal information extracted from depth images. Then, passing these feature vectors to Multi-class Support Vector Machine (MSVM) to perform the action classification task. The proposed representation of the human action ensures it is invariant to the scale of the subjects/objects and the orientation to the camera, while it maintains the correlation among different body parts. The experimental results demonstrate the superior performance of the proposed approach to the state-of-the-art methods.

In the future, more action categories will be included in more diverse and complicated movements. Hence, more complex activities can be explored to exploit the effectiveness of the proposed technique. These activities should incorporate more subjects to improve recognition in a cross subject test. Also, we will extend the proposed method to be implemented in other real-world applications.

References

[1] Rodrigo Cilla, Miguel A. Patricio, Antonio Berlanga, and José M. Molina, "Human Action Recognition with Sparse Classification and Multiple-View Learning," In Expert Systems Journal, Wiley Publishing Ltd, August 2013.

[2] Alexandros André Chaaraoui, Pau Climent-Pérez, and Francisco Flórez-Revuelta, "A Review on Vision Techniques Applied to Human Behaviour Analysis for Ambient-Assisted Living" In the International Journal of Expert Systems with Applications, Volume 39, Issue 12, September 2012.

[3] Heng Wang, Cordelia Schmid, "Action Recognition with Improved Trajectories," In Proceedings of the 2013 IEEE

International Conference on Computer Vision (ICCV '13), Sydney, Australia, pp. 3551-3558, December 2013.

[4] Ronald Poppe, "A Survey on Vision-Based Human Action Recognition," In Image and Vision Computing Journal, Volume 28, Issue 6, pp 976-990, June 2010.

[5] C. Lakshmi Devasena, R. Revathí, and M. Hemalatha, "Video Surveillance Systems – A Survey," In the International Journal of Computer Science Issues (IJCSI), Volume 8, Issue 4, No 1, pp. 635-642, July 2011.

[6] Arnold Wiliem, Vamsi Madasu, Wageeh Boles, and Prasad Yarlagadda, "An Update-Describe Approach for Human Action Recognition in Surveillance Video," In Proceedings of the 2010 International Conference on Digital Image Computing: Techniques and Applications (DICTA '10), Sydney, Australia, pp. 270-275, December 2010.

[7] Guangyu Zhu, Qingming Huang, Changsheng Xu, Liyuan Xing, Wen Gao, and Hongxun Yao, "Human Behavior Analysis for Highlight Ranking in Broadcast Racket Sports Video," In IEEE Transactions on Multimedia, Volume 9, Issue 6, pp. 1167-1182, October 2007.

[8] Mehreen Mumtaz, and Hafiz Adnan Habib, "Evaluation of Activity Recognition Algorithms for Employee Performance Monitoring," In the International Journal of Computer Science Issues (IJCSI), Volume 9, Issue 5, No. 3, pp. 203-210, September 2012.

[9] Yan-Ching Lin, Min-Chun Hu, Wen-Huang Cheng, Yung-Huan Hsieh, and Hong-Ming Chen, "Human Action Recognition and Retrieval Using Sole Depth Information," In Proceedings of the 20th ACM International Conference on Multimedia (MM'12), Nara, Japan, pp. 1053-1056, November 2012.

[10] Lulu Chen, Hong Wei, James Ferryman, "A Survey of Human Motion Analysis Using Depth Imagery," In Pattern Recognition Letters, Volume 34, Issue 15, pp. 1995–2006, November 2013.

[11] Alexandros Iosifidis, Anastasios Tefas, and Ioannis Pitas, "Multi-view Human Action Recognition: A Survey," In Proceedings of the 2013 Ninth International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP '13), Beijing, China, pp. 522-525, October 2013

[12] Hong-Min Zhu and Chi-Man Pun, "Human Action Recognition with Skeletal Information from Depth Camera," In Proceedings of the IEEE International Conference Information and Automation (ICIA), pp. 1082 – 1085, Yinchuan, China, August 2013

[13] Vennila Megavannan, Bhuvnesh Agarwal, and R. Venkatesh Babu, "Human Action Recognition using Depth Maps," In proceedings of the International Conference on Signal Processing and Communications (SPCOM), Bangalore, India, pp. 1-5, July 2012.

[14] Jamie Shotton , Andrew Fitzgibbon , Mat Cook , Toby Sharp, Mark Finocchio , Richard Moore , Alex Kipman , and Andrew Blake, "Real-Time Human Pose Recognition in Parts from Single Depth Images," In Communications of the ACM, Volume 56, Number 1, pp.116-124, January 2013.

[15] Alexandros Andre Chaaraouia, José Ramón Padilla-Lópeza, Pau Climent-Pérezb, and Francisco Flórez-Revuelta, "Evolutionary Joint Selection to Improve Human Action Recognition with RGB-D Devices," In the International Journal of Expert Systems with Applications, Volume 41, Issue 3, pp. 786–794, February 2014.

[16] Jungong Han, Ling Shao, Dong Xu, and Jamie Shotton, "Enhanced Computer Vision with Microsoft Kinect Sensor: A Review" In IEEE Transactions on Cybernetics, Volume 43, Issue 5, pp. 1318 - 1334, October 2013.

[17] Michalis Raptis, Darko Kirovski, and Hugues Hoppe, "Real-Time Classification of Dance Gestures from Skeleton Animation" In Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA '11), ACM: Vancouver, British Columbia, Canada, pp. 147-156, August 2011.

[18] Hong-Min Zhu and Chi-Man Pun, "Human Action Recognition with Skeletal Information from Depth Camera," In Proceedings of the IEEE International Conference Information and Automation (ICIA), Yinchuan, China, pp. 1082 – 1085, August 2013.

[19] Yun Han, Sheng-Luen Chung, Jeng-Sheng Yeh, and Qi-Jun Chen, "Localization of RGB-D Camera Networks by Skeleton-based Viewpoint Invariance Transformation," In the International IEEE Conference on Systems, Man, and Cybernetics (SMC'13), Manchester, United Kingdom, pp. 1525-1530, October 2013.

[20] Xiaodong Yang, Chenyang Zhang, and YingLi Tian, "Recognizing Actions Using Depth Motion Maps-Based Histograms of Oriented Gradients," In Proceedings of the 20th ACM International Conference on Multimedia (MM '12), New York, USA, pp. 1057-1060, November 2012.

[21] Xiaodong Yang, and Yingli Tian, "EigenJoints-Based Action Recognition Using Naïve-Bayes-Nearest-Neighbor" In Proceeding of the International IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Providence, Rhode Island, USA, pp. 14-19, June 2012.

[22] Wee Hong Ong, Takafumi Koseki, and Leon Palafox, "Unsupervised Human Activity Detection with Skeleton Data from RGB-D Sensor," In Proceedings of the 2013 Fifth International Conference on Computational Intelligence, Communication Systems and Networks (CICSYN '13), Madrid, Spain, pp. 30-35, June 2013.

[23] Yang Zhao, Zicheng Liut, Lu Yang, and Hong Cheng, "Combing ROB and Depth Map Features for Human Activity Recognition" In Proceedings of the Signal & Information Processing Association Annual Summit and Conference (APSIPA ASC), Hollywood, Canada, pp. 1-4, December 2012.

[24] Xisheng He, Zhe Wang, Yingbin Zheng, and Xiangyang Xue, "A Simplified Multi-Class Support Vector Machine with Reduced Dual Optimization" In Pattern Recognition

Letters Journal, Volume 33, Issue 1, pp. 71-82, January 2012.

[25] Xiaowei Yang, Qiaozhen Yu, Lifang He, and Tengjiao Guo, "The One-Against-All Partition Based Binary Tree Support Vector Machine Algorithms for Multi-Class Classification," In the Neurocomputing Journal, Volume 113, pp. 1-7, August 2013.

[26] Henry Joutsijoki, and Martti Juhola, "Kernel Selection in Multi-Class Support Vector Machines and its Consequence to the Number of Ties in Majority Voting Method," In Artificial Intelligence Review Journal, Volume 40, Issue 3, pp. 213-230, October 2013.

[27] Yann Guermeur, "A Generic Model of Multi-Class Support Vector Machine," In the International Journal of Intelligent Information and Database Systems, Volume 6, Issue 6, pp. 555-577, October 2012.

[28] Wanqing Li, Zhengyou Zhang, and Zicheng Liu, "Action Recognition Based on a Bag of 3D Points," In Proceedings of the IEEE International Computer Vision and Pattern Recognition Workshops (CVPRW), San Francisco, CA, pp. 9-14, June 2010.

[29] http://research.microsoft.com/en-us/um/people/zliu/ActionRecoRsrc/default.htm, Last access: July 2014.