A COMPARATIVE STUDY ON HUMAN ACTION RECOGNITION USING MULTIPLE SKELETAL FEATURES AND MULTICLASS SUPPORT VECTOR MACHINE

Saiful Islam[1], Mohammad Farhad Bulbul[2*], Md. Sirajul Islam[3]

[2]Jessore University of Science and Technology, Bangladesh
[1,3]Bangabandhu Sheikh Mujibur Rahman Science and Technology University, Bangladesh

ABSTRACT

This paper proposes a framework for human action recognition (HAR) by using skeletal features from depth video sequences. HAR has become a basis for applications such as health care, fall detection, human position tracking, video analysis, security applications, etc. We have used joint angle quaternion and absolute joint position to recognition human action. We also mapped joint position on $SE(3)$ Lie algebra and fuse it with other features. This approach comprised of three steps namely (i) an automatic skeletal feature (absolute joint position and joint angle) extraction (ii) HAR by using multi-class Support Vector Machine and (iii) HAR by features fusion and decision fusion classification outcomes. The HAR methods are evaluated on two publicly available challenging datasets UTKinect-Action and Florence3D- Action datasets. The experimental results show that the absolute joint position feature is the best than other features and the proposed framework being highly promising compared to others existing methods.

KEYWORDS

Human Action Recognition, Absolute Joint Position, Joint Angle Quaternion, SE (3) Lie Algebra Absolute Pair & Support Vector Machine

1. INTRODUCTION

Human Action Recognition (HAR) is one of the most important and challenging research areas of computer vision today. HAR is used widely for the purpose of real life applications, including intelligent video surveillance, to detect dangerous events, monitor people living alone, video analysis, assistive living, robotics, telemedicine, health care, content-based video search, video-game, human-computer interaction [1-2]. The goal of this work is to recognize human actions by using the human skeleton feature in realistic videos, such as movies, videos on the internet and surveillance videos. By using depth sensors, like Microsoft Kinect (Figure 1) or other similar devices nowadays, it is possible to design action recognition systems exploiting depth maps. Microsoft Kinect is also a good source of information because depth maps are not affected by environment light variations, provide body shape, and simplify the problem of human detection and segmentation. Again, research on human action recognition has initially focused on learning and recognizing activities from video sequences which is captured by conservative RGB cameras. On the other hand, the recent publication of cost-effective 3D depth cameras using structured light or time-of-flight sensors, there has been great interest in solving the problem of human action recognition by using 3D data. Here, it is noted that compared with traditional color

DOI:10.5121/mlaij.2018.5201

images, depth images (Figure 2 shows action sequences of depth images) are unaffected in lighting conditions. According to Yang *et al.*[3], color and texture is unacceptable in the depth images which make the tasks of human detection and segmentation easier than the other process. However, according to Shotton *et al.*[4], human skeleton information can be extracted from depth images which provide additional information for action recognition. The Kinect sensor is a motion sensing device and its name is a combination of kinetic and connects. It is originally designed as a natural user interface (NUI) for the Microsoft Xbox 360. Two users can be detected by Microsoft at the same time and their skeletons in 3D with 20 joints representing body junctions like the feet, knees, hips, shoulders, elbows, wrists, head, etc., are computed. In this work, we will use these skeletal features (such as absolute joint position, joint angle), by the Microsoft Kinect sensor. There are a lot of datasets available such as MSR-Action 3D dataset in Li *et al.*[5], UTKinect-Action dataset in Xia *et al.*[6], Florence3D-Action dataset in Seidenari *et al.*[7] etc. Our job is specific to the actions being performed by the human participants.

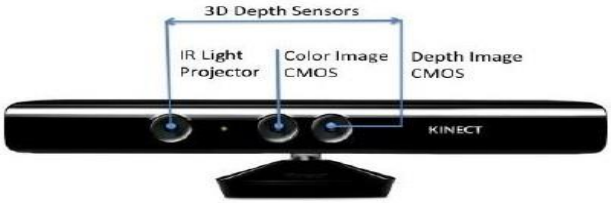

Figure 1. Microsoft Kinect camera.

Figure 2. Example of a depth video sequence for *Tennis Serve* action.

Recognizing actions from videos though, has been extensively researched upon over the past few decades but still, it is way behind the actual deployment to real applications. Since, human activities in videos are not constrained, and there is an abundance of noise like unstable motion, varied range of background, pose diversity etc., human action recognition is a tricky problem. There are numerous reasons to why HAR still remains an open problem. One of the key problems in action recognition is camera motion. Since for recognizing actions, capturing of motion has been the most important cue, any noisy motion may impose a hindrance. In realistic settings, a camera keeps on moving, which in turn results in non-constant background and even the subject of action changing forms in the same scenario. Even with minimalistic camera motion, categorizing actions undergoes severe other challenges, like large inter class variability. Apart from all these, visual appearance of a human subject may differ to a large extent. There may be differences in poses, locations, camera orientations, and light settings. And in certain cases, either the human subjects or the objects may be occluded, which makes it difficult for actions to be detected. There are some other challenges in classification of videos. Among of these one type of problem is intra-class variability, and inter-class similarity. The rest of the sections in this work are organized as: in section 2, we are briefly go through all the primary work in the field of action recognition that has led us here, where we will discuss all the state-of- the-art techniques, their strengths and weaknesses. In section 3, we will explain about the human action recognition and categorization. Skeletal feature extraction methods and classification methods are also discussed in this section. In section 4, the result of our work is discussed in this section. Also action classification from different dataset and the experimental result are also presented in this section. Finally, in section 5 concludes the work.

2. RELATED WORK

Various types of sensor data have been used for human action recognition. Early work on action recognition was done with RGB image and video data. Aggarwal *et al.*[8] took a comprehensive survey focused on human action recognition in images and videos, and described and compared the most widely used approaches. Action recognition and monitoring using inertial sensor data was explored by Maurer *et al.*[9]. However, these sensors are more intrusive and expensive than RGB cameras, and also can't be set up and accessed as easily as them. Going beyond RGB images and videos, the recent advent of inexpensive RGB-D sensors such as the Microsoft Kinect that directly provide depth information has triggered an increasing interest in using depth images for pose estimation in Ganapathi *et al.*[10]. Various types of depth image data approaches have been used for HAR. A method based on depth images and temporal ordering of unique poses was presented by Sheikh *et al.*[11]. Depth images coupled with RGB images have more information available to discriminate different poses and actions, compared to RGB images. Bulbul *et al.*[12] proposed a method where three different kinds of discriminative features were fused for depth video sequences to tackle with the action recognition problem. They computed three feature descriptors employing DMMs, CT-HOG, LBP and EOH. DMMs were utilized to capture specific appearances and shapes in a depth video sequence. In another work [13], a novel feature descriptor based on Depth Motion Maps (DMMs), Contourlet Transform (CT), and Histogram of Oriented Gradients(HOGs) was proposed to classify human actions from depth video sequences. Firstly, CT was implemented on DMMs, then HOGs were computed for each contourlet sub-band. Finally, for the depth video sequence HOG features used as feature descriptor. Again in [14], they proposed another feature extraction scheme for the real-time human action recognition from depth video sequences was taken by using Local Binary Patterns (LBPs), and Edge Oriented Histograms (EOHs) in another work. Here, LBPs were calculated within overlapping blocks to capture the local texture information, and the Edge Oriented Histograms (EOHs) were computed within non-overlapping blocks to extract dense shape features. Li *et al*. [5] used a sampling method to sample from 3D points in depth images to recognize actions. Wang *et al.* [15] extracted Random Occupancy Patterns from depth sequences, used sparse coding to encode these features, and classified actions. Chen *et al.* [16] summarized research on pose estimation and action recognition using depth images. A filtering method to extract STIPs from depth videos (called DSTIP) was discussed by Xia *et al.* [17] which was effectively suppressing the noisy measurements. Moreover, to describe the local 3D depth cuboid around the DSTIPs with an adaptable supporting size, a novel depth cuboid similarity feature (DCSF) was built in that work. Another, HAR method via coupled hidden conditional random field's model was expressed by Liu *et al.*[18]. Also, both RGB and depth sequential information were fused in that work. Wang *et al.*[19] used dense trajectories representation to represent interest points in each frame and tracked them based on the information from a dense optical wield. They also develop a novel descriptor which is robust to camera motion and can be used in more realistic environments. Liu *et al.*[20] presented another approach based on key-frame selection and pyramidal motion feature representation. They used this representation to select the most informative frames from a large pool of action sequences. Ding *et al.*[21] proposed a method to learn the low dimensional embedding with a manifold functional variant of principal component analysis (mfPCA). Fletcher *et al*.[22] developed a method of principal geodesic analysis, a generalization of principal component analysis to the manifold setting.

Skeleton-based human action recognition can be classified into two main categories: joint-based approaches and body part-based approaches. Joint-based approaches consider human skeleton as a set of points, whereas body part-based approaches consider human skeleton as a connected set of rigid segments. Since joint angles measure the geometry between connected pairs of body parts of human skeleton so joint angles can be classified as part-based approaches. Hussein *et al.*[23] represented human skeletons by using the 3D joint locations, and the joint trajectories were

modeled using a temporal hierarchy of covariance descriptors. A similar representation used by Lv *et al.*[24] with Hidden Markov Models (HMMs). Sheikh *et al.*[11] used a set of 13 joint trajectories in a 4-D XYZT to represent a human action, and affine projections were compared using a subspace angles-based view-invariant similarity measure. A human skeleton was represented using pairwise relative positions of the joints by Wang *et al.*[25], and the temporal evolutions of this representation were modeled using a hierarchy of Fourier coefficients. Furthermore, an actionlet based approach used by Oreifej *et al.*[26], where discriminative joint combinations were selected using a multiple kernel learning approach. Body parts and joint coordinates extracted from depth images in that work. After that they had used the histogram of oriented 4D normal of body parts and joint coordinates to recognize actions. Sunj *et al.*[27] combined both the 3D skeleton information and depth images to detect and recognize human actions. Their feature set for a static pose has about 700 elements, including joint positions and Histogram of Oriented Gradients (HOG) of RGB and depth images. Ellis *et al.*[28] presented algorithms to reduce latency at the time of recognizing actions. Theodorakopoulos *et al.*[29] proposed a method for action recognition. To obtain robust and invariant pose representations, skeletal data were initially processed. Devanne *et al.*[30] proposed a framework to extract human action captured through a depth sensor. This proposed solution was capable to capture both the shape and the dynamics of the human body simultaneously. In this work, final classification was completed by usingkNN. Since spatio-temporal features and skeleton joints features are complementary to each other, Zhu *et al.*[31] discussed another feature-level fusion of these two features in by using random forests method. To represent skeletal motion in a geometric, a process was presented by Salma *et al.*[32] whose observability matrix was characterized as an element of a Grassmann manifold. A robust informative joints based HAR method was proposed by Jiang *et al.*[33] (2015). They also analyzed the mean contributions of human joints for each action class via differential entropy of the joint locations. After extracting the 3D skeletal joint locations from depth images, Gan *et al.*[34] computed APJ3D from the action depth image sequences by employing the 3D joint position features and the 3D joint angle features. They recognized actions by using random forests with employing improved Fourier Temporal Pyramid.

3. INTRODUCE RECOGNITION METHOD

In this section, we firstly take a look at the Support Vector Machine (SVM) shortly. We firstly take a look at the data acquisition by Microsoft Kinect, special Euclidean Group $SE(3)$, and Support Vector Machine shortly. Then, the proposed framework will be discussed comprehensively. A framework of this action recognition process is showed in Figure 3.

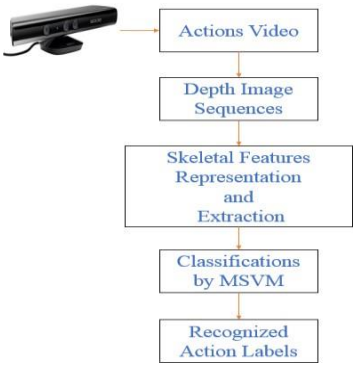

Figure 3.A framework of our human action recognition process.

A number of classification techniques have been successfully applied to recognized actions e.g. HMMs, k-NN, ANNs, DTW, CNNs, Boosting, and SVMs. Over the last years, SVMs is the most popular classification technique used in action recognition in videos.

3.1 Data acquisition by Microsoft Kinect

One of the major components of the Kinect sensor is its ability to infer human motion by extracting human silhouettes in skeletal structures. It extracts the skeletal joints of a human body as 3D points using the Microsoft SDK. It provides a skeleton model with 20 joints as shown in Figure 4. To detect human subjects, the availability of depth information helps researchers to implement simpler identification procedures. The advantages of this technology, with respect to classical video-based ones in Chen *et al.*[16]:

a) Being less sensitive to variations in light intensity and texture changes
b) Providing 3D information by a single camera
c) Maintaining privacy, it is not possible to recognize the facial details of the people captured by the depth camera. This feature helps to keep identity confidential.

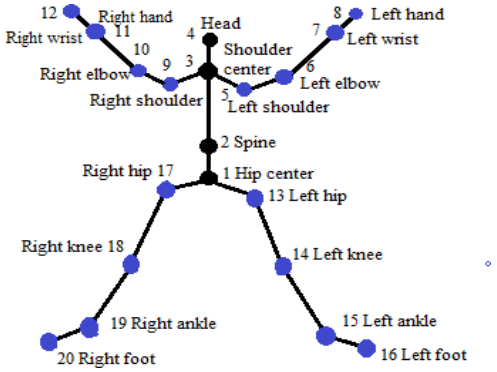

Figure 4. Skeleton joints detected by Microsoft SDK according as Sheikh *et al.*[11].

This Kinect camera can produce a skeletal data over an image frame with 20 important joints in the model of human body as shown in Figure 4. These joints are bone parts, in which the basic fixation bones are points of "hip center" joint (node 1), "spine" joint (node 2), "shoulder center" joint (node 3) and "head" joint (node 4); the positions of the flexible bones are "wrist left" joint (node 7), "hand left" joint (node 8), "wrist right" joint (node 11), "hand right" joint (node 12), "ankle left" joint (node 15), "foot left" joint (node 16), "ankle right" joint (node 19) and "foot right" joint (node 20); the joints of the moving bones are the positions of "shoulder left" joint (node 5), "elbow left" joint (node 6), "shoulder right" joint (node 9), "elbow right" joint (node 10), "hip left" joint (node 13), "knee left" joint (node 14), "hip right" joint (node 17) and "knee right" joint (node 18). In particular, each 3D data of human bone joints obtained from the camera denotes with three coordinates (x, y, z)called (Horizontal, Vertical, and Depth) for a joint position. In this work, each joint position I is described as the transpose matrix with three values in a coordinate (x, y, z) and expressed as follows:

$$\left.\begin{array}{l} I_1 = [x_1 \quad y_1 \quad z_1]^T \\ I_2 = [x_2 \quad y_2 \quad z_2]^T \\ I_3 = [x_3 \quad y_3 \quad z_3]^T \\ \quad \ldots \ldots \ldots \\ \quad \ldots \ldots \ldots \\ \quad \ldots \ldots \ldots \\ I_n = [x_n \quad y_n \quad z_n]^T \end{array}\right\} \quad (1),$$

where n describes the number of the joint points, $n = 1, 2, 3, \dots\dots\dots, 20$. In building 3D data, each image frame is determined to be 20 bone joints and each joint is in the 3D coordinate(x, y, z). Therefore, this frame can be described as a column vector I_n of 60 variables. This vector is arranged from matrices in equation (1) and each frame is expressed as follows:

$$P_1 = [I_1 \quad I_2 \ \dots\dots\dots\dots\dots \ I_n]^T \tag{2},$$

Assume that one video clip has *k* frames, $k = 1, 2, 3, .,$ with human activities is defined as the following matrix, $\boldsymbol{P}$:

$$\boldsymbol{P} = [P_1 \quad P_2 \quad P_3 \dots\dots \ P_k] \tag{3},$$

This matrix **P** can is also expressed another way as follows:

$$\boldsymbol{P} = \begin{bmatrix} P_{1,1} & P_{1,2} \dots\dots\dots\dots & P_{1,k} \\ P_{2,1} & P_{2,2} \dots\dots\dots\dots & P_{2,k} \\ P_{3,1} & P_{3,2} \dots\dots\dots\dots & P_{3,k} \\ & \dots\dots\dots\dots\dots\dots\dots & \\ & \dots\dots\dots\dots\dots\dots\dots & \\ P_{n,1} & P_{n,2} \dots\dots\dots\dots & P_{n,k} \end{bmatrix} \tag{4},$$

3.2 Special Euclidean group $SE(3)$

Rigid body rotations and translations in 3D space are members of the special Euclidean group $SE(3)$ which is a matrix Lie group. Hence, we have represented the relative geometry between a pair of body parts as a point in $SE(3)$. The entire human skeleton as a point in the Lie group $SE(3) \times \dots\dots\dots \times SE(3)$. Here, direct product between Lie groups was shown by the notation $\times$. We will refer to the readers Hall *et al.* [35] for a general introduction to Lie groups and Murray *et al.* [36] (1994) for further details on $SE(3)$ and rigid body kinematics. The special Euclidean group, denoted by $SE(3)$, is the set of all 4 by 4 matrices of the form

$$P(R, \quad \vec{d}\,) = \begin{bmatrix} R & \vec{d} \\ 0 & 1 \end{bmatrix} \tag{5},$$

where $\vec{d} \in R^3$, and $R \in R^{3\times3}$ is a rotation matrix. Members of $SE(3)$ act on points $z \in R^3$ by rotating and translating them, we get:

$$\begin{bmatrix} R & \vec{d} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} z \\ 1 \end{bmatrix} = \begin{bmatrix} Rz + d \\ 1 \end{bmatrix} \tag{6},$$

By the usual matrix multiplication, elements of this set interact. This can be smoothly organized to form a curved 6 dimensional manifold from a geometrical point of view, giving them the structure of a Lie group. The 4 by 4 identity matrix I_4 is a member of $SE(3)$ and is referred to as the identity element of this group. The tangent plane to $SE(3)$ at the identity element I_4 is known as the Lie algebra of $SE(3)$, and is denoted by se(3). It is a 6 dimensional vector space formed by all 4 by 4 matrices of the form$\begin{bmatrix} U & \vec{w} \\ 0 & 0 \end{bmatrix}$, where $\vec{w} \in R^3$ and U is a 3 by 3 skew-symmetric matrix. For any element

$$B = \begin{bmatrix} U & \vec{w} \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & -u_3 & u_2 & w_1 \\ u_3 & 0 & -u_1 & w_2 \\ -u_2 & -u_1 & 0 & w_3 \\ 0 & 0 & 0 & 0 \end{bmatrix} \in se(3) \tag{7},$$

And its vector representation $vec(B)$ is given by

$$vec(B) = [u_1, u_2, u_3, w_1, w_2, w_3] \quad (8),$$

3.3 Support vector machine

S**upport vector machines** (**SVMs**) are supervised learning algorithms which are also known as **support vector networks**. Vladimir N. Vapnik and Alexey Ya. Chervonenk were invented the original SVM algorithm in 1963. SVMs are the most prominent machine learning algorithms that analyze data and recognize patterns. This algorithm is also one of the most robust and accurate Machine Learning methods which has a sound theoretical foundation and efficient training algorithms. Depending on the nature of the data, such a separation might be linear or non-linear. Let us consider a linear classifier (or, hyperplane)

$$f(X) = w^T x + b \quad (10),$$

where w represents a weight vector, x is the input feature vector and b represents the position of the hyperplane. Here,

(a) if the input vector is 2-dimensional, the linear equation will represent a straight line.

(b) if the input vector is 3-dimensional, the linear equation will represent a plane.

(c) if input vector more than 3D, the linear equation will represent a hyperplane.

That is, if $x_1$is a unknown vector (Figure 6) which we want to classify, then

$$class(x_1) = \begin{cases} C_1 & if\ w^T x_1 + b > 0 \\ C_2 & if\ w^T x_1 + b < 0 \end{cases} \quad (11),$$

And x_1 lies on the hyperplane when $w^T x_1 + b = 0$.

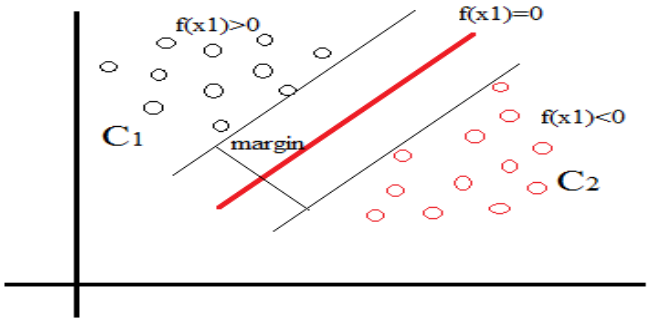

Figure 5. Two classes are separated by a hyperplane.

According to Nguyen *et al.* [37], the SVM algorithm, the linear hyperplane is an area to divide the data set into two subsets collection according to the linear hyperplane. Assume that pattern elements are expressed as follows:

$$(x_1,\ y_1), (x_2,\ y_2), (x_3,\ y_3) \dots\dots\dots, (x_m,\ y_m) \quad (12),$$

where $x_i \in R_n$ while $y_i \in \{-1, +1\}$ is subclass of x_i. Therefore, one needs to find a plane making Euclidean distance between two layers if we want to identify the hyperplane. In particular, a

vector with the distance values close to the plane is called the support vector, in which, the positive value is $y = +1$ or H1 and the negative one is in the region of $y = -1$ or H2. The SVM algorithm is to find an optimal hyperplane for classification of two classes and expressed as follows:

$$f(x) = w^T x + b \quad (13),$$

Assume that the equation of hyperplane is $w^T x + b = 0$, where w is the vector with perpendicular points to the separating hyperplane and with $w \in R_n$ and $\frac{|b|}{||w||}$ is distance from this hyperplane to the origin, and $||w||$ is the magnitude of w. Moreover, d_+ (d_-) is the shortest distance from the hyper-boundary to positive (negative) samples. Also, the region bounded by these two hyperplanes is called the margin of this hyperplane. The distance between $w.x + b = +1$ and $w.x + b = -1$ is the margin of this hyperplane (Figure 6). By applying the distance rule between two straight lines, we get the margin, $m = \frac{2}{||W||}$. Suppose that all training data satisfy the following constraints:

$$w.x_i + b \geq +1, \text{ in which } y_i = +1 \quad (14),$$

$$w.x_i + b \geq -1, \text{ in which } y_i = -1 \quad (15),$$

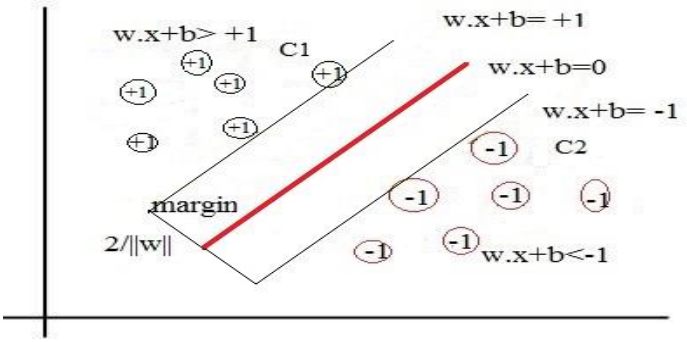

Figure 6. Classification of an unknown feature vector with the help of hyperplane.

From two equations (12) and (13), one has:

$$y_i(w.x_i + b) \geq +1 \quad (16a),$$

and $$y_i(w.x_i + b) = 1 \quad (16b),$$

where x_i is a support vector where support vector is the input vectors that just touch the boundary of the margin. Simply, support vectors are the data points that lie closest to the decision surface (or hyperplane).

4. EXPERIMENTAL RESULTS

There are different types of datasets available to use action recognition process such as MSR-Action3D datasets, UTKinect-Action Dataset, Florence3D-Action Dataset, KTH Dataset, WEIZMANN Dataset, Hollywood Dataset, Multi-camera Human Action Video Data etc. In this work we have used UTKinect-Action and Florence3D-Action datasets. Action and Florence3D-Action datasets.

4.1 UTKinect-Action and Florence3D-Action datasets setup

The UTKinect-Action dataset was collected as part of research work on action recognition from depth sequence. This dataset was captured by using a stationary Kinect sensor with Kinect for Windows SDK. It consists of 10 actions performed by 10 different subjects. Where each subject performed every action in two times. Altogether, there are 199 action sequences. The 3D locations of 20 joints are provided in this dataset. The dataset collected at the University of Texas's. These 10 actions of UTKinect-Action Dataset are: **walk, stand up, sit down, pick up, carry, throw, push, pull, wave hands, clap hands.** Three channels were recorded: RGB, depth and skeleton joint locations. Again, the Florence3D-Action dataset was collected at the University of Florence during 2012. This dataset was captured by using a stationary Kinect sensor. It consists of 9 actions by performing 10 different subjects. Each subject performed every action two or three times. Altogether, there are 215 action sequences. Its 9 actions are: **wave, drink from bottle, answer phone, clap, tight lace, sit down, stand up, read watch, bow.** Only 15 joints are provided in Florence3D-Action dataset on 3D. Due to high intra-class variations, this dataset is challenging than UTKinect-Action dataset.

4.2 Experimental Results and Comparison of UTKinect-Action Dataset

In this section we will evaluate the experimental results based on joint angle quaternion, $SE(3)$ Lie Algebra Absolute Pair, and absolute joint position of UTKinect-Action datasets. Our proposed method's performance will be compared with other competitive methods. Again, after completing our experiment based on joint angle quaternions feature, we get 94.95% recognition accuracy. A comparison of results with other previous methods is given in Table 1.

Table 1: Recognition results comparison of various skeletal representations with joint angle quaternions feature of UTKinect-Action dataset.

Author	Method	Accuracy Rates
Zhu *et al.* (2013) [31]	RF	91.9%
Xia *et al.* (2013) [17]	DCSF	90.1%
Xia *et al.* (2013) [17]	HMMS	90.92%
Devanne *et al.* (2014) [30]	*k*NN	91.5%
Joint Angle Quaternions	MSVM	94.95%

If we notice [Table 2] the class wise accuracy of these 10 actions of UTKinect-Action dataset, we will get a few misclassifications.

Table 2: Class wise accuracy of UTKinect-Action dataset by using joint angle quaternions.

Actions	Walk	Sit down	Stand up	Pick up	Carry	Throw	Push	Pull	Wave hand	Clap hand
Classifications Rates	90	100	100	90	88.89	90	90	100	100	100

From the confusion matrix in Figure 8 (a confusion matrix, also known as an error matrix is a specific table layout that allows visualization of the performance of an algorithm) we notice that the 'Sit-down' action classification rate is 100% that means this classification method can classify this action accurately.

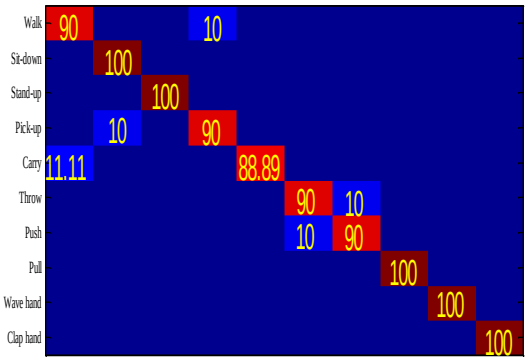

Figure 7: Confusion matrix of joint angle quaternion feature.

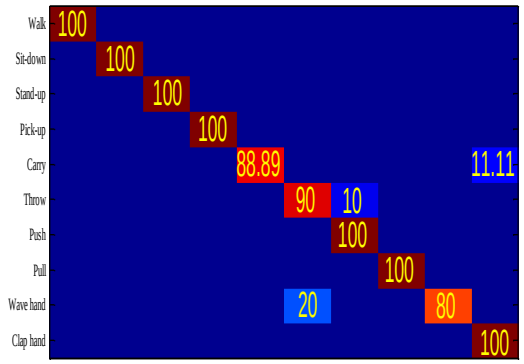

Figure 8: Confusion matrix of $SE(3)$ Lie algebra absolute joint pair feature.

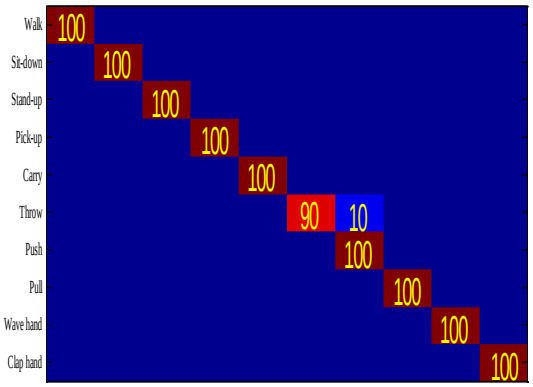

Figure 9: Confusion matrix of absolute joint positions feature.

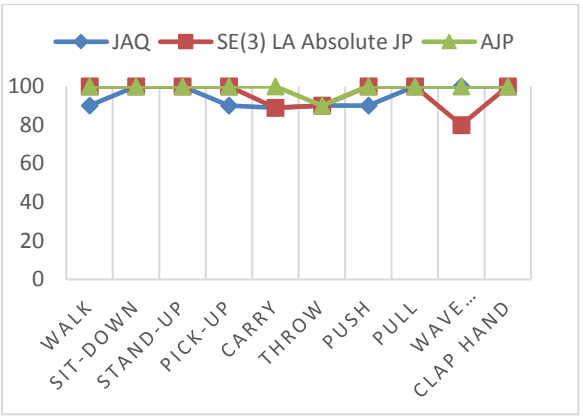

Figure 10: Result comparison between joint angle quaternion, SE (3) Lie algebra absolute pair, and absolute joint position features.

Also, many other actions such as 'Stand-up', 'Pull', 'Wave hand', and 'Clap hands' are accurately classify by this classification method. But rest of the action of UTKinect-Action dataset can't fully classify.

Table 3: Recognition rates comparison for various skeletal representation with $SE(3)$ Lie algebra absolute pair feature.

Authors	**Method**	**Accuracy Rates**
Fletcher *et al.* (2012) [22]	LARP+PGA	91.26%
Gan *et al.* (2013) [34]	APJ3D and RF	92%
Theodorakopoulos *et al.* (2014) [29]	Boost Performance	90.95%
Jiang *et al.* (2015) [33]	Linear CRFs	91.5%
$SE(3)$ Lie Algebra Absolute Pair	MSVM	95.96%

We also notice that due to inter-class similarity there are a number of misclassifications that means some actions are confused with other action in this data set. For example, due to inter class similarity for carry action there is 11.11% misclassification that means the action carry is confused with the action walk. Similarly, due to inter class similarity for push action there is 10% misclassification that means the action 'Push' is confused with the action 'Throw'.That means, the feature joint angle quaternions is not as stronger as we need for this UTKinect-Action

dataset. To improve the accuracy rate we may apply feature fusion process. After completing our second experiment by using $SE(3)$ Lie algebra absolute pair we get the recognition rate (Table3) 95.96%.Moreover, if we notice [Table 4] the class wise accuracy of these 10 actions of UTKinect-Action dataset by using $SE(3)$ Lie algebra absolute pair feature, we will get also a few misclassifications.

Table 4: Class wise accuracy of UTKinect-Action dataset by using $SE(3)$ Lie algebra absolute pair feature.

Actions	Walk	Sit down	Stand up	Pick up	Carry	Throw	Push	Pull	Wave hand	Clap hand
Classifications Rates	100	100	100	100	88.89	90	100	100	80	100

Again, if we observe Figure 8, we will notice the minimization of misclassification rate. 'Walk' and 'Pick-up' actions were misclassified by joint angle quaternion feature but by using $SE(3)$ Lie algebra absolute pair feature these misclassifications are removed. Furthermore, another feature called absolute joint position is applied on UTKinect-Action dataset in our work. At this time, recognition accuracy rate is increased than the previous two features which is 98.99%.

Table 5: Recognition rates comparison for various skeletal representation with absolute joint positions feature on UTKinect-Action dataset.

Authors	**Method**	**Accuracy Rates**
Liu *et al.* (2012) [20]	Coupled hidden conditional RF	92%
Ding *et al.* (2015) [21]	STFC	91.5%
Salma *et al.* (2015) [32]	Linear SVM	88.5%
Absolute Joint Position	MSVM	98.99%

From Figure 9, we notice that most of the actions are accurately classified except the action 'Throw'. So, it is clear that the raw feature absolute joint positions are suitable for UTKinect-Action datasets. In Figure 10, a comparison of class wise action accuracy between joint angle quaternion, $SE(3)$ Lie algebra absolute joint pair, and absolute joint position features is given. From this comparison we can say that absolute joint position feature shows the better performance than the previous two features.

4.3 Experimental Results and Comparison of Florence3D-Action Dataset

In similar manner, when we complete our experiment by using joint angle quaternions, absolute joint positions, and $SE(3)$ Lie algebra absolute joint pair features on Florence3D-Action dataset, we get another type of results (Table 9). Due to inter-class similarity among these actions, recognition rate is quite less than the previous dataset.

Table 6: Results comparison for our three skeletal features on Florence3D-Action datasets.

Authors	**Method**	**Accuracy Rates**
Fletcher *et al.* (2004) [22]	LARP+PGA	79.01%
Seidenari *et al.* (2013) [7]	NBNN	82%
Ellis *et al.* (2013) [28]	CRF	65.7%
$SE(3)$ Lie Algebra Absolute Pair	MSVM	75.45%
Joint Angle Quaternions	MSVM	84.55%
Absolute Joint Position	MSVM	81.82%

From the confusion matrix (Figure 12) of $SE(3)$ Lie algebra absolute joint pair feature based classifications, a number of misclassifications are observed. Only two actions are accurately classified.

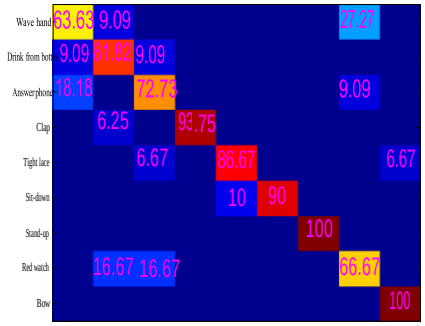

Figure 11: Confusion matrix of joint angle quaternion feature.

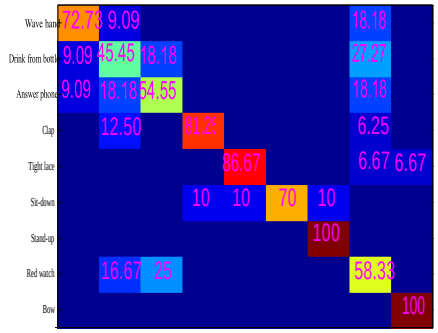

Figure 12: Confusion matrix of $SE(3)$ Lie algebra absolute joint pair feature.

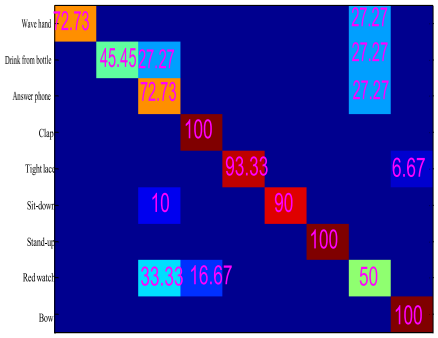

Figure 13: Confusion matrix of absolute joint position feature.

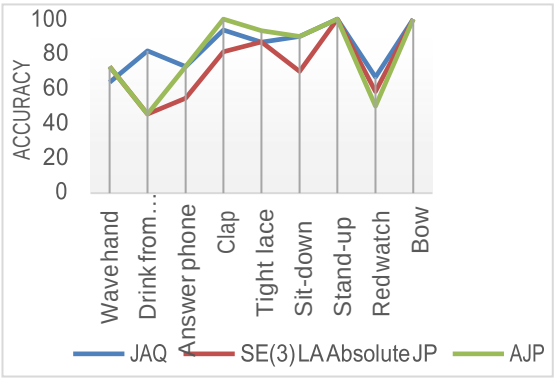

Figure 14: Result comparison between joint angle quaternion, SE (3) Lie algebra absolute joint pair, and absolute joint position features.

From Table 6, we notice that the action classification accuracy by the derived feature $SE(3)$ Lie algebra absolute pair is 75.45% but this feature can't classify all action properly except the actions 'Bow', and 'Stand Up' (Figure 12). Again, if we notice Table 6, it will be clear that these actions accuracy is better than the SE (3) Lie algebra absolute joint pair feature. Also another action 'Clap' among these actions is fully classified (Figure 13). A comparison of accuracy results is given in Figure 14.So, by applying feature fusion strategy of $SE(3)$ Lie algebra absolute joint pair with absolute joint positions, we will be able to minimize the misclassification of these actions. After feature fusion (Figure 15) of absolute joint position and $SE(3)$ Lie algebra absolute pair features we will get accuracy rate be 81.82% and the number of misclassifications will be reduced.

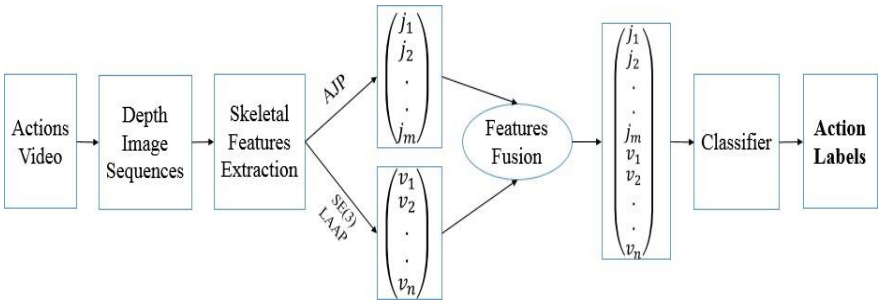

Figure 15. Absolute joint position (AJP) and $SE(3)$ Lie algebra absolute joint pair ($SE(3)$ LAAP) features fusion framework.

5. CONCLUSIONS

Human action recognition is one of the main areas of research nowadays. In this work, we have shown skeleton based techniques for HAR. This approach is done with the help of skeletal data. Skeleton joint positions and skeletal joint angle from the Kinect sensor are collected from the inertial sensor. In our first experiment, we have evaluated the action recognition results by using joint angle quaternions of human skeleton. But there appear a few misclassifications of some actions. Then to develop our accuracy rate, we have introduced another two skeletal feature known as $SE(3)$ Lie algebra absolute joint pair and absolute joint positions. After that, we have applied a feature fusion method on absolute joint positions features with $SE(3)$ Lie algebra absolute joint positions features. Finally, we get a better result than the first approach and compare with different skeletal features based classifications. The proposed system have recognized action via multi-class support vector machine (MSVM). Experiments carried out on two datasets: UTKinect-Action dataset and Florence3D-Action dataset. When we compare with other skeletal-based solution our approach show competitive performance than others previous methods. Finally, we have observed that our proposed features are more appropriate on UTKinect-Action dataset than Florence-3D Action dataset.

References

[1] Theodoridis, T., Agapitos, A., Hu, H. & Lucas, S.M. (2008) Ubiquitous robotics in physical human action recognition: a comparison between dynamic ANNs and GP. In: ICRA, pp. 3064– 3069.

[2] Chen, C., Jafari, R. & Kehtarnavaz, N. (2015) Improving human action recognition using fusion of depth camera and inertial sensors. IEEE Trans. Hum.-Mach. Syst. 45(1), 51–61.

[3] Yang, X. & Tian, Y. (2014). Super normal vector for activity recognition using depth sequences. In: CVPR, pp. 804–811.

[4] Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R. & Blake, A. (2011). Real-time human pose recognition in parts from single depth images. In:CVPR, pp. 1297–1304.

[5] Li, W., Zhang, Z. & Liu, Z. (2010). Action recognition based on a bag of 3D points. In 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pages 9-14.

[6] Xia, L., Chen, C.C. & Aggarwal, L.K. (2012). View invariant human action recognition using histograms of 3D joints. In 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pages 20–27.

[7] Seidenari, L., Varano, V., Berretti, S., Bimbo, A. D., & Pala, P. (2013). Recognizing Actions from Depth Cameras as weakly Aligned Multi-part Bag-of-Poses. In CVPRW.

[8] Aggarwal, J. K. & Ryoo, M. S. (2011) Human activity analysis: A review. ACM Computing Survey, 43(3):16.

[9] Maurer, U., Smailagic, A., Siewiorek, D. P. & Deisher. M. (2006). Activity recognition and monitoring using multiple sensors on different body positions. In International Workshop on Iarable and Implantable Body Sensor Networks, pages 113–116.

[10] Ganapathi, V., Plagemann, C., Koller, D. & Thrun, S. (2012). Real-time human pose tracking from range data. In European Conference on Computer Vision, pages 738–751.

[11] Sheikh, Y., Sheikh, M. & Shah. M. (2005). Exploring the Space of a Human Action. In ICCV.

[12] Bulbul, M. F., Jiang, Y, & Ma, J. (2015) DMMs-based multiple features fusion for human action recognition. IJMDEM 6(4): 23-39.

[13] Bulbul, M. F., Jiang, Y, & Ma, J. (2015) Real-time Human Action Recognition Using DMMs-based LBP and EOH Features. ICIC (1)2015: 271-282.

[14] Bulbul, M. F., Jiang, Y, & Ma, J. (2015) Human Action Recognition Based on DMMs, HOGs and Contourlet Transform. IEEE International Conference on Multimedia Big Data, pages 389–394.

[15] Wang, J., Liu, Z., Chorowski, J., Chen, Z. and Wu, Y. (2012). Robust 3d action recognition with random occupancy patterns. In European Conference on Computer Vision (2), pages 872–885.

[16] Chen, L., Ii, H. & Ferryman, J. M. (2013). A survey of human motion analysis using depth imagery. Pattern Recognition Letters, 34:1995–2006.

[17] Xia, L. & Aggarwal, J. K. (2013). Spatio-Temporal Depth Cuboid Similarity Feature for Activity Recognition Using Depth Camera. CVPR '13 Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition. pages 2834-2841.

[18] Liu, A.-A., Nie, W.-Z., Su, Y.-T., Ma, L., Hao. T. & Yang, Z.-X. (2015). Coupled hidden conditional random fields for RGB-D human action recognition, Signal Processing, vol. 112, pp. 74–82.

[19] Wang, H., Klaser, A., Schmid, C. & Liu, C-L. (2011). Action recognition by dense trajectories. In Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, pages 3169-3176. IEEE.

[20] Liu, L., Shao, L. & Rockett, P. (2012). Boosted key-frame selection and correlated pyramidal motion-feature representation for human action recognition. Pattern Recognition.

[21] Ding, W., Liu, K., Cheng, F. & Zhang, J. (2015). STFC: spatio-temporal feature chain for skeleton-based human action recognition, Journal of Visual Communication and Image Representation, vol. 26, pp. 329–337.

[22] Fletcher, P. T., Lu, C., Pizer, M. & Joshi, S. C. (2004). Principal geodesic analysis for the study of nonlinear statistics of shape. IEEE Transactions on Medical Imaging, 23(8):995– 1005, August 2004. 1, 3, 6, 7.

[23] Hussein, M., Torki, M., Gowayyed, M & El-Saban, M. (2013). Human Action Recognition Using a Temporal Hierarchy of Covariance Descriptors on 3D Joint Locations. In IJCAI.

[24] Lv, F. & Nevatia. R. (2006). Recognition and Segmentation of 3D Human Action Using HMM and Multi-class Adaboost. In ECCV.

[25] Wang, J., Liu, Z., Wu, Y. & Yuan, J. (2012). Mining actionlet ensemble for action recognition with depth cameras. In Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, pages 1290–1297.

[26] Oreifej, O., & Liu, Z. (2013). HON4D: Histogram of oriented 4D normal for activity recognition from depth sequences. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 716–723, 2013.

[27] Sung, J., Ponce, C., Selman, B. & Saxena, A. (2012). Unstructured human activity detection from RGBD images. In IEEE International Conference on Robotics and Automation, pages 842–849.

[28] Ellis, C., Masood, S.Z., Tappen, M.F., Jr., J.J.L., Sukthankar, R. (2013). Exploring the trade-off between accuracy and observational latency in action recognition. International Journal of Computer Vision, 101(3), 420 436, DOI: 10.1007/s11263-012-0550-7.

[29] Theodorakopoulos, I., Kastaniotis, D., Economou, G. & Fotopoulos, S. (2014). Pose-based human action recognition via sparse representation in dissimilarity space. Journal of Visual Communication and Image Representation, vol. 25, no. 1, pp. 12–23.

[30] Devanne, M., Wannous, H., Berretti, S., Pala, P., Daoudi, M. & Bimbo, A.D. (2014). 3D Human Action Recognition by Shape Analysis of Motion Trajectories on Reimannian Manifold. IEEE Transactions on System Man and Cybernetics.

[31] Zhu, Y., Chen, W. & Guo, G. (2013). Fusing Spatiotemporal Features and Joints for 3D Action Recognition, CVPRW.

[32] Salma, R., Wannous, H., Daoudi, K. & Srivastava, A. (2015). Accurate 3D action recognition using learning on the Grassmann manifold. Pattern Recognition, 48(2):556 – 567.
[33] Jiang, M., Kong, J., Bebis, G. & Huo, H. (2015). Informative joints based human action recognition using skeleton contexts. Signal Processing: Image Communication, vol. 33, pp. 29– 40.
[34] Gan, L., & Chen, F. (2013). Human action recognition using APJ3D and random forests. Journal of Software, vol. 8, no. 9, pp. 2238– 2245.
[35] Hall, B. (2013). Lie Groups, Lie Algebras, and Representations: An Elementary Introduction. Springer.
[36] Murray, R. M., Li, Z. & Sastry. S. S. (1994). A Mathematical Introduction to Robotic Manipulation. CRC press.
[37] Nguyen, T. H. & Trinh, H. (2016). An PCA-SVM Algorithm for Classification of Skeletal Data-Based Eigen Postures. American Journal of Biomedical Engineering 2016, 6(5): 147-158 DOI: 10.5923/j.ajbe.20160605.03.

Authors

Saiful Islam recently completed his Masters of Science in Applied Mathematics in 2017 and Bachelor of Science degrees in Mathematics in 2015 from Bangabandhu Sheikh Mujibur Rahman Science and Technology University, Gopalganj, Bangladesh. He is interested at working in Signal Processing, Image Processing, Computer Vision, Pattern Recognition, Machine Learning, and Artificial Intelligence.

Dr. Md. Farhad Bulbul is currently working as an Assistant Professor in the Department of Mathematics at Jessore University of Science and Technology, Jessore, Bangladesh. He received his PhD degree from the Department of Information Science, **Peking University**, Beijing, China, in 2016. His PhD research direction was "Statistical Learning and Intelligent Information Processing". He achieved the *Peking University President's award* for his outstanding performance in doctoral study (Including course work and research).
From 2012 to 2016, he held the CGS doctoral fellowship from Chinese Government. He has published papers in Thomson Reuters' ESCI Indexed journal, IEEE International Conference Proceedings (EI Indexed) and Lecture Notes in Computer Science (Springer-Verlag, EI Indexed). His research focuses on Computer Vision, Deep Learning, Pattern Recognition, and Image Processing.

Md. Sirajul Islam is currently working as an Assistant Professor in the Department of Mathematics at Bangabandhu Sheikh Mujibur Rahman Science and Technology University, Gopalganj-8100, Bangladesh. He has published papers in Physical Science International Journal. His research focuses on Image Processing, Machine Learning, Wavelet Analysis, Optimization, Differential Equations, Numerical Analysis, Finite Difference Method, Finite Element Method, and Fluid Dynamics.