IOP Conference Series: Materials Science and Engineering

Research on Action Recognition Method Based on Weighted DTW Algorithm

To cite this article: Zhen Zhang and Ya Zhang 2019 *IOP Conf. Ser.: Mater. Sci. Eng.* **646** 012057

IOP Conf. Series: Materials Science and Engineering **646** (2019) 012057 doi:10.1088/1757-899X/646/1/012057

Research on Action Recognition Method Based on Weighted DTW Algorithm

Zhang Zhen[1, *] and Zhang Ya[1]

College of Mechatronic Engineering , North University of China , Taiyuan 030051, China

[*]E-mail: zbzz2012@163.com

Abstract: Action recognition technology is an important part of artificial intelligence. In order to improve the rate of action recognition, this paper designs an action recognition method based on Weighted DTW algorithm. Firstly, the Kinect2.0 is used to obtain the three-dimensional data of the human joint points. Then the quaternion method is used to define the action sequences. The weight of joint is calculated according to the participation in different types of action. Then the improved DTW algorithm is used to design the action recognition experiment. The experimental results show that the action recognition algorithm designed in this paper has better recognition rate and timeliness than traditional DTW algorithm and F-DTW algorithm.

1. Introduction

With the rapid development of the artificial intelligence industry, action recognition technology has received more and more attention[1]. Human action recognition technology can be widely used in many fields including public place situation recognition, medical and motion monitoring, and new human-computer interaction[2]. However, the human body has the features of high mobility, huge individual differences and disorder[3]. In the process of action recognition, the camera is affected by light and shadow changes, visual angle, and picture clarity. This series of random factors makes the study of human action recognition very complicated[4].

Depth sensors have developed rapidly in recent years. And the problem that the precision is insufficient when we use two-dimensional images to study action recognition is gradually solved. Kinect2.0 is a relatively mature depth sensor in recent years. Compared with the previous generation, Kinect2.0 adopts the TOF ranging principle, which is almost immune to illumination changes and textures when acquiring external action information[5]. This undoubtedly provides great help for the study of human action recognition.

The algorithm research of action recognition is of great significance for the development of human body gesture recognition. Samsu Sempena[6] et al. used a dynamic time warping method to compare video input to a defined action list. This method requires a lot of calculations when analyzing complex actions performed by multiple people ,which will reduce the efficiency of recognition. T. Vajda[7] proposed a motion recognition method based on fast dynamic time warping and feedforward neural network. This approach speeds up the identification process by introducing constraints that shorten the length of the time series. However, when the test sequence differs greatly from the action template, its recognition accuracy will decrease.

In this paper, we propose a weighted calculation method based on the participation of joint points

IOP Conf. Series: Materials Science and Engineering **646** (2019) 012057 doi:10.1088/1757-899X/646/1/012057

in different action sequences, and realize the recognition of action sequences using the Weighted DTW algorithm in the experiment. Experimental results show that the algorithm has good recognition accuracy and efficiency.

2. Human action description

2.1. Feature vector establishment

In the process of realizing human action recognition, it is necessary to establish an action model as a template for identifying objects. The human skeleton obtained by the Kinect 2.0 sensor contains the pose information of the joint point[8]. As figure 1 and figure 2 show, the whole structure of skeleton continues from SpineBase as the root joint to head and extremities. In the process of identification, it is not enough to set up the action model by simply obtaining the joint point data. Converting the corresponding joint point data into action features that can be recognized is the basis for implementing action recognition.

Considering that the activity of spine changes minimally when making different movements, we use SpineBase as the reference point to establish the action feature, and calculate the joint direction data from the data flow acquired by the sensor. We define a continuous sequence of action by creating multiple joint-direction data features included in multiple frames as a set of feature vectors.

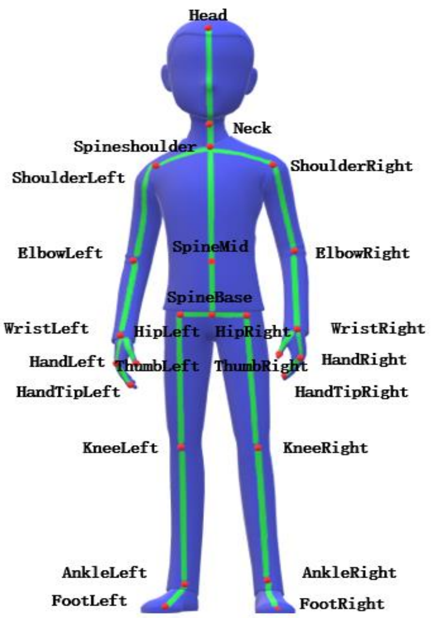

Figure 1. Skeleton map recognized by Kinect2.0.

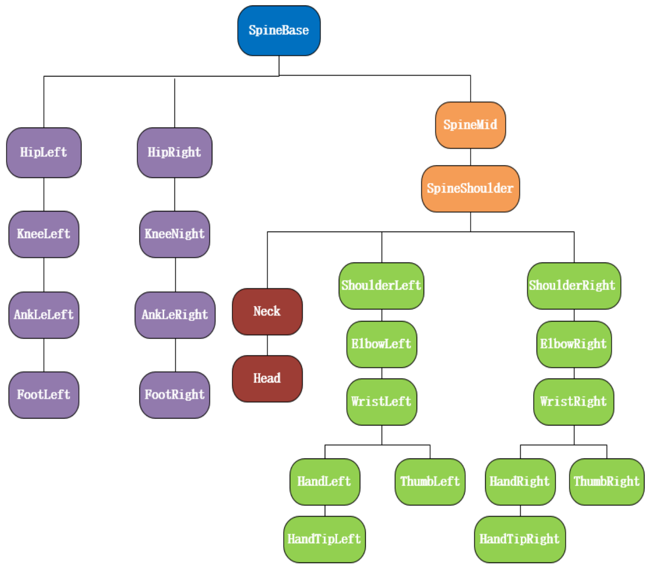

Figure 2. The joint hierarchical structure

2.2. Quaternion representation

A quaternion is a complex number with three imaginary parts, which is expressed as follows:

$$q = xi + yj + zk + w \tag{1}$$

We choose quaternions to describe the joint pose state. On the one hand, compared to Euler angles, the quaternion can avoid the gimbal lock phenomenon when the degree of freedom is insufficient caused by coincidence between two relatively rotating plane. On the other hand, quoting a quaternion can reduce the amount of calculation and the space occupied by the storage of calculation[9].When using kinect2.0 to get the joint data stream, we can call the IBody::GetJointOrientations function to get the quaternion part which contains the actions information as shown in figure 3.

We can calculate the distance between quaternions from the acquired data. Figure 4 is a schematic diagram of calculating the quaternion distance. q_1, q_2 are two unit quaternions. S_1 rotates to S_2 by $q_2q_1^{-1}$.If $q_2q_1^{-1}=[w,x,y,z]$, then define the distance between q_1 and q_2 as:

$$d(q_1,q_2)=\arccos w \tag{2}$$

We can define the method of action recognition by using $d(q_1,q_2)$ which indicates the direction and distance difference of q_1 to q_2.

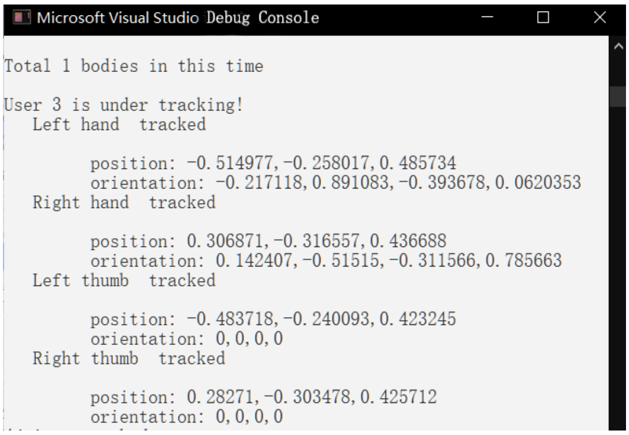

Figure 3. Quaternions data acquisition.

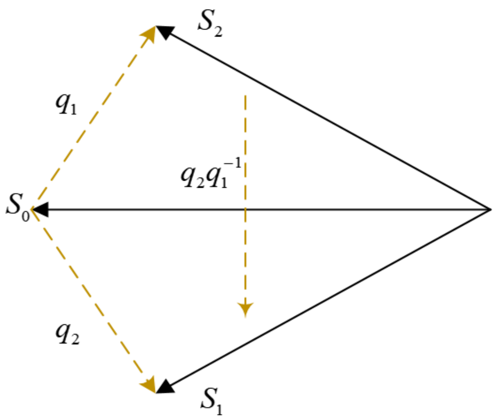

Figure 4. Quaternions distance calculation.

3. Research on action recognition algorithm

Considering that we have used data changes of multiple joint direction to define the action sequence, to implement action recognition is the process of identifying and matching for a sequence of consecutive actions that have been defined. The action process is time-ordered. In the process of action recognition, the sequence of actions of the recognition target often differs from the sequence of action templates over time due to the individual differences and the randomness of the recognition object. Therefore, it is necessary to use the corresponding algorithm to achieve target matching in the research of action recognition.

3.1. Standard Dynamic time warping

The Dynamic time warping algorithm can be effectively applied to the identification match between two action sequences. The main idea of DTW is to extend and compress two time series, and obtain the minimum warping path between the two sequences in dynamic matching[10].

Define two sequences of actions, where F is the template sequence and T is the test sequence, with lengths m and n:

$$\mathrm{F}=(f_1,f_{2,}\cdots f_{\mathrm{i}}\cdots f_m) \tag{3}$$

$$T=(t_1,t_{2,}\ldots t_{\mathrm{j}}\ldots t_n) \tag{4}$$

f_i and t_{j} are sets of joint direction data related to the i th frame and the j th frame of the action sequences F and T, respectively. Usually m≠n, therefore, we use the dynamic programming method to create an $m\times n$ matrix grid. The elements (i,j) in the matrix, which can be expressed in Euclidean Metric , represent the similarity between the eigenvalue f_i of the i th frame in the template sequence and the eigenvalue t_{j} of the j th frame in the test sequence.

$$d(f_i,t_j)=(f_i-t_j)^2 \tag{5}$$

We use the DTW method to adjust a curve with the highest similarity from w_1 to w_k in the square, which is the warping path as shown in figure 5. Define the warping path as W, and the element g in W is $w_g = (i, j)_g$, then:

$$W = w_1, w_2, w_3 \ldots w_g \ldots w_k \tag{6}$$

$$\max(m,n) \le k \le m+n-1 \tag{7}$$

W satisfies the minimum matching distance between time series F and T:

$$DTW(F,T) = \min\left\{K^{-1}\left(\sum_{k=1}^{K} w_k\right)^{1/2}\right. \tag{8}$$

K is the compensation coefficient for the planning path of different lengths.

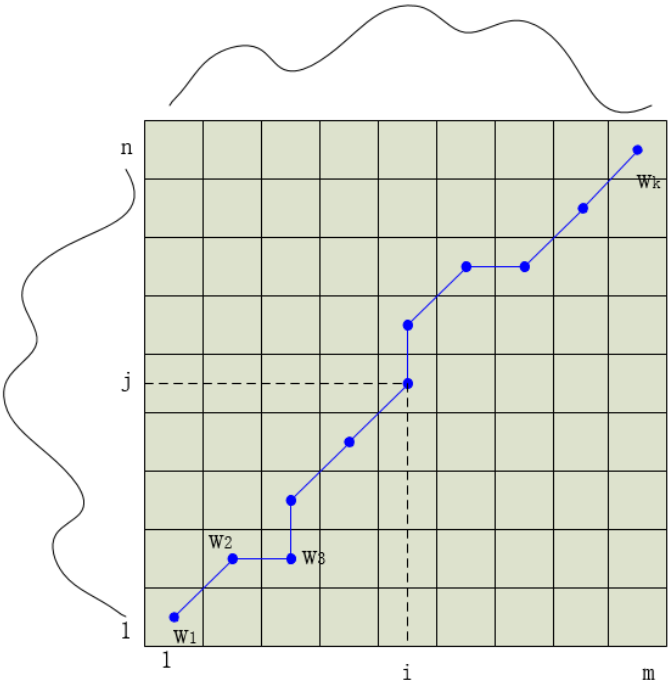

Figure 5. The best planning path by DTW

Meanwhile, We needs to meet three constraints as below.

3.1.1. Boundary conditions. The order should be specified from A to B regardless of the length of the sequence of actions, which means the planning path is from the bottom left to the top right.

3.1.2. Continuous conditions. It is forbidden to cross a point in the sequence to match. The next point in the path $w_g = (i_g, j_g)$ need to satisfy $i_g - i_{g-1} \le 1$ and $j_g - j_{g-1} \le 1$ if $w_{g-1} = (i_{g-1}, j_{g-1})$. In this way we can ensure that each of the match is aligned with an adjacent point, and all the points in the sequence participate in path regularization.

3.1.3. Monotonic conditions. The sequence of actions should be arranged in order. So the path needs to be planned according to the monotony of time.

Restricted by the three conditions[11], the warping path has only three directions to plan in the square[12], and the next grid point is limited to $(i+1, j)$, $(i, j+1)$ and $(i+1, j+1)$. From the above constraints, a cumulative distance can be defined to represent the similarity between the two sequences.

$$L(i, j) = d(f_i, t_j) + \min\{\gamma(i-1, j), \gamma(i, j-1), \gamma(i-1, j-1)\} \tag{9}$$

3.2. F-DTW algorithm

F-DTW restricts and abstracts the traditional DTW algorithm. First, it abstracts the data from the original sequence when warping the path, which is called coarse-grained. Then, in the process of coarse-grained warping path, it executes the DTW method to plan the path again, which is called fine-grained[13]. This algorithm reduces the search space, so the algorithm recognition complexity can be reduced and the system response time can be improved in sequence identification. However, when the time sequence is coarsely granulated and then expanded to a fine-grained path, the calculation accuracy of the DTW distance will be reduced.

3.3. Weighted DTW algorithm

The traditional DTW algorithm needs to calculate and match each feature vector of each frame in the action sequence in the process of warping path[14]. When calculating the total matching distance[15], the pose change of each joint point needs to be considered in the sequence matching. However, in the process of action recognition, the participation of each joint is different, and the amount of motion variation of the same joint in different action sequences is also different. In order to reduce the complexity of sequence recognition and improve the timeliness of action recognition, this paper assigns weights according to the participation degree of joint in different actions, and use the weighted DTW algorithm to realize identification matching between the template sequence and the test sequence.

We set the weight coefficient ω_k according to the degree of participation of the joint i in the action sequence. Define the function $Tru(f_i, k)$, $\beta = \left|d\left(f_{i-1}^k, f_i^k\right)\right|$, where f_i^k represents the quaternion representation of the direction data of joint k which is contained in the i th frame of the sequence f . The function satisfies the following relationship:

$$Tru(f_i, k) = \begin{cases} 1, & \beta \neq 0 \\ 0, & \beta = 0 \end{cases} \tag{10}$$

Then the weight coefficient ω_k can be expressed as:

$$\omega_k = \frac{1}{N-1}\sum_{i=1}^{N} Tru(f_i, k) \tag{11}$$

We can acquire the weighted cost function from the improved DTW algorithm:

$$imp(f_i, d_j) = \sum_{k=1}^{K} \omega_k \left|d\left(f_i^k, t_j^k\right)\right| \tag{12}$$

Test different action templates to get a gray-scale value matrix which represents the value of weight coefficient. From the gray-scale value in the square, it can be intuitively seen that the weight coefficient of joints is different in different action templates as shown in figure 6.

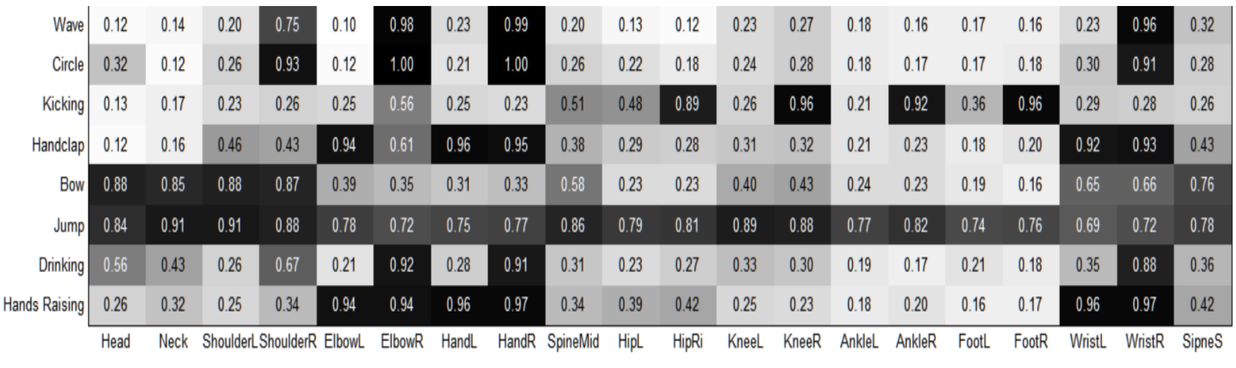

Figure 6. Representation of weight coefficients in squares.

4. Experiment

4.1. Experimental environment
The experiments in this paper were implemented in the Win10 operating system. The hardware environment includes a Kinect2.0 senor and a computer with the Intel i57300-Q CPU. We chose Visual Studio 2017 as the integrated development environment, and use SDK Browser v2.0 (Kinect for Windows) to provide API development interface. The action recognition interface is based on WPF, and the program is written in C#.

4.2. Experimental process
The experimental process is shown in figure 7. First, we use Kinect2.0 sensor to get human motion information to train action samples and build action libraries. Then we use quaternions to represent the sequence of actions and calculate the weight coefficient of joints. The traditional DTW method and the weighted DTW method are used to match the test action sequence and the template action sequence to obtain the comparison results.

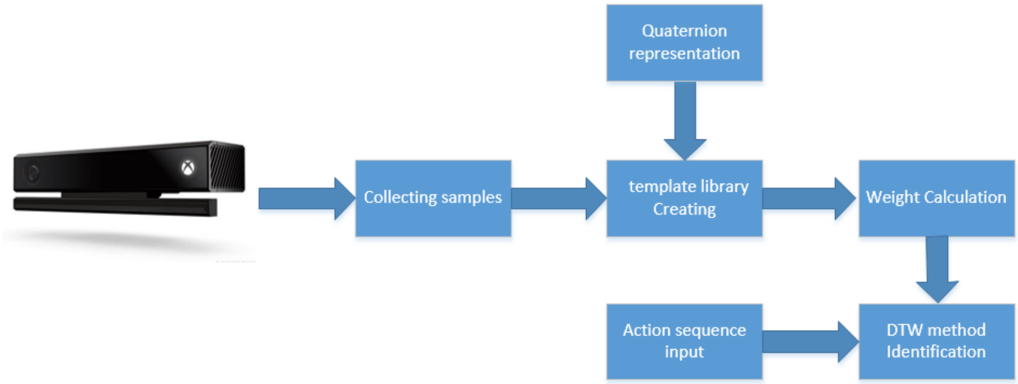

Figure 7. Experimental process using Weighted DTW method

4.3. Experimental operation

4.3.1. Template recording. We used Kinect2.0 to collect motions with a sampling frequency of 15 frames per second as shown in figure 8. 10 test subjects with different individual states were selected and each person performed 20 times on one action. Template recording is completed in WPF project. Click the Motion Capture button in the project to complete the corresponding action before the countdown ends. Repeat this process to complete the template library.

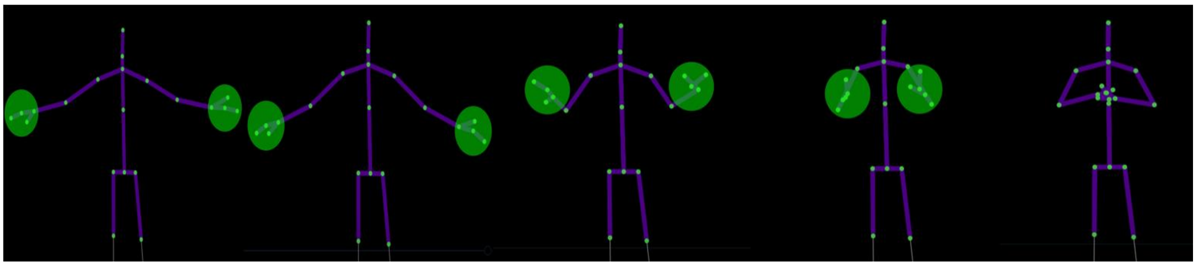

Figure 8. The action sequence of “Handclap”.

4.3.2. Identification test. Click the Load Action Template button in the project to enter the recognition mode. As shown in figure 9-10,the tester performs the template action and then the system begins to identify the action sequence based on the Standard DTW algorithm, F-DTW algorithm and Weighted

DTW algorithm.

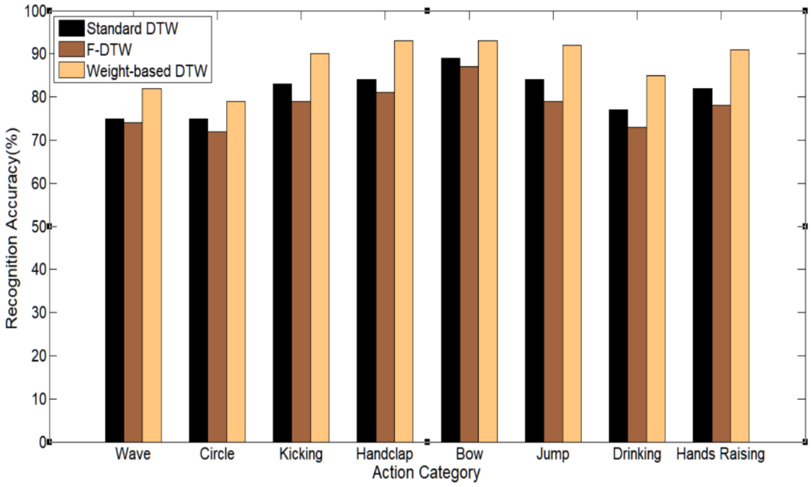

Figure 9. The recognition accuracy of Weighted DTW algorithm and other algorithms.

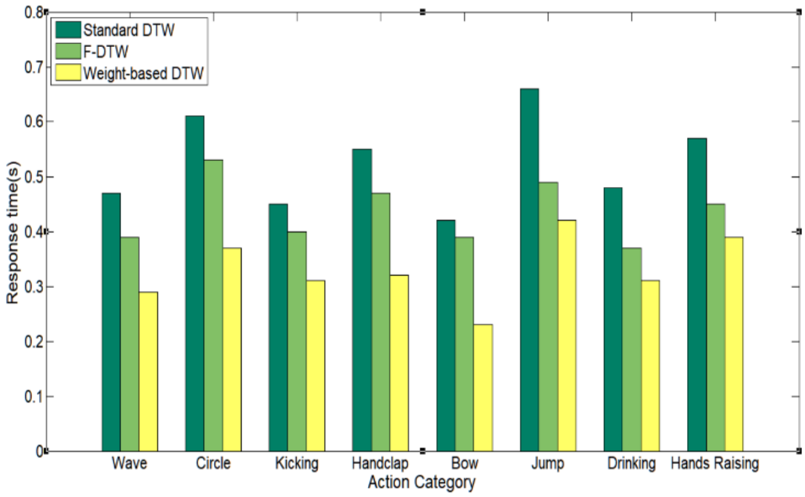

Figure 10. The response time of Weighted DTW algorithm and other algorithms.

4.4. Analysis of experimental results
In the recognition test of the eight action types, the Weighted DTW algorithm proposed in this paper shows a higher recognition rate and a shorter matching time than the Standard DTW Algorithm and the F-DTW Algorithm. Some of the actions in the experiment were not recognized because the Kinect sensor was not ideal for denoising when acquiring joints data. Overall, the Weighted DTW has shown satisfactory accuracy and timeliness in the recognition and action sequences matching.

5. Conclusion and future work
This paper proposes an action definition method using quaternion distances and a Weighted DTW algorithm based on the participation of joints. In the action description process, in order to reduce the calculation and the space occupied by the storage of calculation, we use quaternion distances to define action sequences. The Weighted DTW algorithm proposed in this paper solves the problem of the timeliness caused by the traversal calculation of the Standard DTW Algorithm and the low recognition accuracy of the F-DTW Algorithm. The experimental results show that the overall recognition result of the Weighted DTW algorithm has better accuracy and timeliness.

However, there are some deviations in the results of action recognition in the test when the experiment is disturbed by external conditions. Therefore, in the future work, we need to consider adding denoising and filtering methods to the improvement measures.

References

[1] V. Thanikachalam and K. K. Thyagharajan, "Human action recognition using accumulated motion and gradient of motion from video," 2012 Third International Conference on Computing, Communication and Networking Technologies (ICCCNT'12), Coimbatore, 2012, pp. 1-6.

[2] J. Wang, Z. Liu, Y. Wu and J. Yuan, "Mining actionlet ensemble for action recognition with depth cameras," 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, 2012, pp. 1290-1297.

[3] S. Cherla, K. Kulkarni, A. Kale and V. Ramasubramanian, "Towards fast, view-invariant human action recognition," 2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, Anchorage, AK, 2008, pp. 1-8.

[4] R. Vemulapalli, F. Arrate and R. Chellappa, "Human Action Recognition by Representing 3D Skeletons as Points in a Lie Group," 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, 2014, pp. 588-595.

[5] X. Tang, A. Song 2018 Human pose estimation and its implementation in scenario interaction system of rehabilitation training, *Chinese Journal of Scientific Instrument*, Vol.**39**, No.11, pp. 195-203

[6] S. Sempena, Nur Ulfa Maulidevi and Peb Ruswono Aryan, "Human action recognition using Dynamic Time Warping," Proceedings of the 2011 International Conference on Electrical Engineering and Informatics, Bandung, 2011, pp. 1-5.

[7] T. Vajda, "Action recognition based on Fast Dynamic-Time warping method," 2009 IEEE 5th International Conference on Intelligent Computer Communication and Processing, Cluj-Napoca, 2009, pp. 127-131.

[8] H. Alabbasi, A. Gradinaru, F. Moldoveanu and A. Moldoveanu, "Human motion tracking & evaluation using Kinect V2 sensor," 2015 E-Health and Bioengineering Conference (EHB), Iasi, 2015, pp. 1-4.

[9] B. Jablonski 2012 Quaternion Dynamic Time Warping, *Transactions on Signal Processing*, Vol. **60**, No. 3, pp. 1174-1183.

[10] X. Ruan and C. Tian, "Dynamic gesture recognition based on improved DTW algorithm," 2015 IEEE International Conference on Mechatronics and Automation (ICMA), Beijing, 2015, pp. 2134-2138.

[11] G. Zhang, H. Yu and L. Xiao, "CEW-DTW: A new time series model for text mining," 2018 International Conference on Information and Communications Technology (ICOIACT), Yogyakarta, 2018, pp. 158-162.

[12] S. S. Jambhale and A. Khaparde, "Gesture recognition using DTW & piecewise DTW," 2014 International Conference on Electronics and Communication Systems (ICECS), Coimbatore, 2014, pp. 1-5.

[13] Y. Jin and Z. Bi, "Power Load Curve Clustering Algorithm Using Fast Dynamic Time Warping and Affinity Propagation," 2018 5th International Conference on Systems and Informatics (ICSAI), Nanjing, 2018, pp. 1132-1137.

[14] Young-Seon Jeong, Myong K. Jeong, Olufemi A. Omitaomu 2011 Weighted dynamic time warping for time series classification, *Pattern Recognition*, Vol.**44**, No.9, pp.2231-2240.

[15] Y. Lee and J. Lee, "Kinematic Constraint Method for Human Gesture Recognition Based on Weighted Dynamic Time Warping," 2016 6th International Conference on IT Convergence and Security (ICITCS), Prague, 2016, pp. 1-5.