Research Article

INTERNATIONAL JOURNAL OF ADVANCED ROBOTIC SYSTEMS

International Journal of Advanced Robotic Systems
January-February 2021: 1–9

DOI: 10.1177/1729881420983219
journals.sagepub.com/home/arx
SAGE

Application of human motion recognition technology in extreme learning machine

Anzhu Miao[1] and Feiping Liu[2]

Abstract

Human motion recognition is a branch of computer vision research and is widely used in fields like interactive entertainment. Most research work focuses on human motion recognition methods based on traditional video streams. Traditional RGB video contains rich colors, edges, and other information, but due to complex background, variable illumination, occlusion, viewing angle changes, and other factors, the accuracy of motion recognition algorithms is not high. For the problems, this article puts forward human motion recognition based on extreme learning machine (ELM). ELM uses the randomly calculated implicit network layer parameters for network training, which greatly reduces the time spent on network training and reduces computational complexity. In this article, the interframe difference method is used to detect the motion region, and then, the HOG3D feature descriptor is used for feature extraction. Finally, ELM is used for classification and recognition. The results imply that the method proposed here has achieved good results in human motion recognition.

Keywords

Human motion recognition, extreme learning machine, feature extraction, classifier

Date received: 17 April 2020; accepted: 4 December 2020

Topic Area: Robot Manipulation and Control
Topic Editor: Antonio Fernandez-Caballero
Associate Editor: Antonio Fernandez-Cabellero

Introduction

Visual information is one of the most important sources of information we know about the world and accepts external feedback. A large amount of data research shows that about 75% of the information received by the brain comes from human vision, and then, the brain processes and analyzes the acquired visual information, so visually acquired information plays an important role in messaging and understanding. With the continuous improvement of modern computer technology and the popularization of video smart devices, video image information in daily life is the most important form of visual information. The analysis and identification of human action behaviors in the field of computer vision have always been a challenging subject. The field of human motion recognition research methods also includes research topics in the fields of digital image information processing, intelligent analysis, pattern recognition, and artificial intelligence. Human motion recognition[1,2] refers to the process of processing and analyzing human action behaviors using a computer, and then, performing a process of identifying and classifying on the basis of this. The research on human–computer interaction has been continuously deepened. The existence of robots can replace many tasks that humans are not capable of. Intelligent human–computer interaction will be the theme of present and future society, and intelligence and practicality are also the basic requirements of human–computer interaction systems. It is one

[1] Sports Department, Guizhou University of Finance and Economics, Guiyang, Guizhou, China
[2] PE Department, Wuhan Institute of Technology, Wuhan, Hubei, China

Corresponding author:
Feiping Liu, PE Department, Wuhan Institute of Technology, Wuhan 430205, Hubei, China.
Email: lfp9095@126.com

of the important directions to realize the intelligentization of the machine by capturing and recognizing human motion and communicating with the robot.

In the field of computer vision, research on human motion recognition is booming. This research has a very wide range of application scenarios.[3] In foreign countries, the earliest Johansson proposed a motion recognition method based on human joint motion,[4] and the research on human motion recognition technology has obtained relatively valuable exploratory research results. For example, O'Rourke and Badler[5] began to study the human body estimation algorithm and introduced a three-dimensional human body model in human vision simulation based on single vision. Nagel[6] proposed the idea of hierarchically defining human behavior, providing a new way of thinking for behavior or motion recognition. Yamato et al.[7] identified the time series of human motion and used the bottom-up feature to combine the hidden Markov model to realize the state-based motion recognition algorithm. Polana and Nelson[8] regard the human behavior process as a repeated change of time and space, that is, the action classification and recognition based on the space–time template. Davis and Bobick[9] described the human motion process using motion-history-image and human motion-energy-image, and proposed a motion recognition algorithm based on time and space segmentation. Gritai et al.[10] identified the motion process of human motion using dynamic time planning and established a three-dimensional human body model of attitude estimation. The Visual Surveillance and Monitoring (VSAM) project[11] was jointly developed by the US Department of Defense and the United University. It is mainly to automatically analyze and identify the special environment of the battlefield to achieve unmanned intelligent monitoring. The Cooperative Distributed Vision project[12] was initiated by the Japanese academic revitalization. It mainly uses the active vision agent (AVA) to realize large-scale real-time monitoring, traffic detection, or 3D video. The ADVISOR project[13] mainly studies the behavior patterns and human–computer interactions of individuals and people, and realizes real-time monitoring of airports, subways, parking lots, and so on. The integrated surveillance of crowded areas for public security (ISCAPS) project[14] uses infrared cameras, optical cameras, biometric instruments, and other devices to track people, vehicles, and other objects to prevent the occurrence of vicious events and realize real-time monitoring of the scene. In addition, MIT CSAIL Labs,[15,16] University of Central Florida Computational Visual Laboratory,[17] Stanford University Vision Lab,[18] University of Southern California Institute of Robotics and Intelligent Systems,[19] and the French INRIA Institute,[20] and so on, have also conducted many related research on motion detection, motion tracking, feature extraction, and motion recognition. The Institute of Computer Science of the Chinese Academy of Sciences[21] has studied human motion and analysis and identification, and has applied some research results to the training programs of sports athletes. The Computer Vision Laboratory of Shanghai Jiaotong University[22] has done a lot of research on image feature extraction and matching, rigid motion analysis, and so on, and has obtained a lot of research results: image processing and pattern recognition laboratory[23] also for image processing research on target recognition and tracking, biometrics and behavior analysis, and published some related technology patents. There are also a large number of research in Beijing Jiaotong University, Chinese Academy of Sciences,[24] and some achievements have been made. Although human motion recognition technology has made great progress, there are still many shortcomings. First, shooting angles, shooting distances, lighting changes, occlusions, and target shadows will make the extraction of human motion features inaccurate and incomplete, which will directly affect the results of motion recognition.[25] Second, the robustness of the motion features is not high enough, and the recognition accuracy is not ideal under the influence of interference factors. Third, in high-speed real-time monitoring scenarios, it must reduce the complexity and execution time of the algorithm to improve real time and accuracy.

Extreme learning machine (ELM) is a competent single hidden-layer feedforward neural network learning algorithm.[26,27] It is not the same as conventional neural network learning methods. When performing this method, it is only necessary to set the number of nodes in the hidden layer of the network, and it is not necessary to adjust the input weight of the network and the offset of the hidden elements. It produces a unique optimal solution, is extremely fast, and achieves high generalization performance, enabling high-speed processing speeds such as recognition systems. Zuoren et al.[28] based on the image segmentation algorithm of ELM and based on the determination of optimal parameters, an image-segmentation algorithm based on ELM was established. The correctness and validity of the algorithm were verified by simulation experiments. It is pointed out that this algorithm can complete the segmentation of the image more quickly, and the image segmentation has fewer isolated points and obvious edges. At the same time, the algorithm greatly shortens the training time of the sample. Songlin et al.[29] applied the multilayer ELM to intrusion detection, the detection false negative rate was as low as 0.48%, and the detection speed was more than six times higher than that of other depth models. Big data network records are better expressed with fewer parameters and have advantages in both detection speed and feature expression of intrusion detection. Xiangming and Jianmin[30] applied the ELM and multisource information to the motor fault intelligent diagnosis and used the trained ELM model as the diagnostic decision classifier to judge the running state of the motor. This method can accurately diagnose the motor. The fault type has the features of fast running speed and good accuracy of fault diagnosis, which satisfies the requirements of online real-time diagnosis of the system. The ELM has achieved good results in all of the

above applications, and it just meets the requirements of human motion recognition. Therefore, the ELM can be applied to the research of human motion recognition.

To settle the dispute that the traditional method is not accurate and time complex in human motion recognition, this article uses the ELM to study the human motion recognition. Here, the interframe difference method is firstly used to detect the motion region. This is to prepare for the subsequent feature extraction. Then, the feature extraction is performed using the HOG3D feature descriptor. Finally, ELM is used to classify and recognize the human motion. Through simulation experiments on human motion data sets, the results show that increasing the number of hidden layer nodes in a certain range can improve the accuracy of recognition. However, too many nodes in the hidden layer are likely to cause overfitting to the training data and increase the computational complexity. In motion recognition, there are some cases of recognition errors due to the size, wear, and different speeds of the human body. In addition, the influence of noise on recognition rate is considered in this article. Under the condition of different signal-to-noise ratio (SNR) from 15 dB to 30 dB, the recognition effect of SVM and ELM is analyzed. The recognition method based on ELM is much better than the recognition method based on SVM and has great advantages. This article has applied the ELM to human motion recognition and achieved good results and has a good reference value in the field of human motion recognition.

Method

The basic task of human motion recognition is to extract and describe the essential characteristics of human motion by computer and realize the recognition of action type, which is essentially a multiclassification problem. Therefore, detecting motion regions, extracting effective and representative motion features, and establishing a classifier with good performance are key issues in identifying motions.

Pretreatment

The key step in the extraction of human body features is to extract the human motion area, and the target detection technology is usually applied to realize the extraction of the human motion area. In this article, the interframe difference method is used for motion region detection. The interframe difference method is to use the difference of the position of the moving target between adjacent frames in the scene, where the background is relatively static, and the difference caused by the target motion is obtained by the gray level of the pixel.

For the moving target, the gray value of the moving area pixel has a larger change than the background, the edge part of the target contour changes most sharply, the internal variation of the target body contour is relatively small, and the static background pixel is almost unchanged. According to this motion characteristic, by constructing the motion intensity accumulation map, the motion region of the motion target over a period of time can be obtained. When a video has T frames, the pixel difference between the frame and the previous frame is calculated for each frame to obtain a pixel variation intensity map d, and the expression is as follows

$$\mathrm{d}(t) = (|I(t) - I(t-1)| + |I(t+1) - I(t)|)/2 \quad (1)$$

where $t \varepsilon (2, T-1)$ represents the video frame number. For a video, after obtaining the pixel intensity map d of each frame, all $T-2$ pixel intensity maps are accumulated to obtain an exercise intensity accumulation graph P, as shown in the following equation

$$P(i,j) = \sum_{t=2}^{T-1} d(i,j;t) \quad (2)$$

By normalizing the pixel values and then binarizing, the background can be filtered out, and the region [$x0$, $x1$, $y0$, $y1$] of the target motion in the video is initially obtained.

The motion intensity accumulation map can obtain all the motion regions of the target by calculating the intensity of the change of the target over a period of time. However, the position coordinates of the target in a certain frame cannot be accurately obtained, that is, the target motion region cannot be detected. To obtain the positional area where each frame of the moving object is located, it is necessary to calculate the pixel variation intensity map d of each frame and one frame before and after. For a frame, the five pixel change intensity maps are accumulated before and after, and then binarized to obtain the target motion region [$xx1$, $xx2$, $yy1$, $yy2$]. The target motion area coordinates [$xx1$, $xx2$, $yy1$, $yy2$] and [$x0$, $x1$, $y0$, $y1$] obtained by the 11 frames are taken as intersections, so that the area where the moving target of each frame is located can be obtained.

Human motion feature extraction

After the target motion area is detected, the human motion feature extraction is required. Image feature is one of the basic properties of an image. Feature extraction is a good part of the image that allows the computer to automatically recognize the target in this image. It is the starting point for analyzing this image. The final performance of a target recognition algorithm is closely related to the selection of features. The main purpose of human motion feature extraction is to extract motion feature data, to extract key feature information, and to describe human action behavior. Local features are used in this article to describe the target. The use of local features to describe the target has two major advantages. First, the feature points of the local features can be

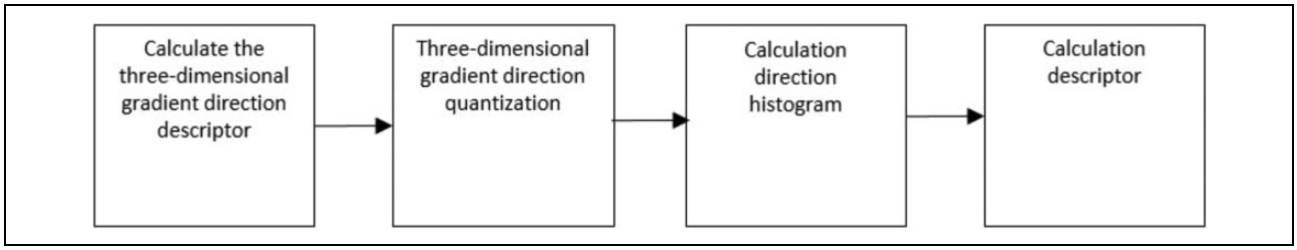

Figure 1. HOG3D feature extraction process.

stably calculated. The local feature points are used to represent the target image, which can significantly reduce a large amount of useless information of the image, and the computational efficiency is greatly improved. On the other hand, when an object is disturbed, some redundant information can recover important information from unobstructed feature points even if it is occluded.

HOG3D feature, a local descriptor, is usually used to represent a part of an image or video. Usually, local area features can also be obtained by spatiotemporal detection. A complete image or video could be described by many features calculated using different scales and coordinates. The extraction process of HOG3D features is shown in Figure 1.

(1) Calculate the 3D gradient descriptor

The first step is to calculate the gradient points of interest and points around the points based on different time and space dimensions. It is common to use time–space pyramids, that is, to calculate gradients in different time-space dimensions. In fact, each time-space dimension video sequence needs to be readjusted.

Given N scales, if $\sigma_{xy}, \sigma_t{}^t$ represent the spatial-temporal scale factors, respectively, the equation for the scale factor is as follows

$$z = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \sigma_{xy}^{-2i} \sigma_t^{-j} \tag{3}$$

To improve the computational productiveness, the operation of the average gradient vector requires the addition of an integral image algorithm. Given that $v(x, y, t)$ represents a video sequence, then $v_{\partial x}, v_{\partial y}$, and $v_{\partial t}$ describe $v(x, y, t)$ in x, y, t partial differentials, and the $v_{\partial x}$-integral video equation is known

$$iv_{\partial x}(x, y, t) = \sum_{x \le x, y \le y, t \le t} v_{\partial x}(x', y', t') \tag{4}$$

In the same way, $iv_{\partial y}, iv_{\partial t}$ are available. Arbitrarily given a three-dimensional cuboid $b = (x, y, t, w, h, l)^T$, the coordinates of the point of interest are (x, y, t^T), where w delegates the width, h delegates the height, and l delegates the length. The average gradient $\bar{g}_b = (\bar{g}_{b\partial x}, \bar{g}_{b\partial y}, \bar{g}_{b\partial t})^T$ is known, as shown in the following equation

$$\begin{aligned} g_{b\partial x} = [&iv_{\partial x}(x + w, y + h, t + l) - iv_{\partial x}(x + w, y, t + l) \\ &+ iv_{\partial x}(x, y, t + l)] - [iv_{\partial x}(x + w, y + h, t) \\ &- iv_{\partial x}(x, y + h, t) - iv_{\partial x}(x + w, y, t) + iv_{\partial x}(x, y, t)] \end{aligned} \tag{5}$$

(2) Three-dimensional gradient direction quantization

The gradient direction histogram can be regarded as a circle. The circle is first divided into N blocks, and each center angle is $2\pi/N$. Assuming a three-dimensional space average gradient vector $\bar{g}_b$, first map $\bar{g}_b$ to the coordinate system, which can be described by multiplication of the matrix, $p = (p_1, \cdots, p_n)^T$ represents the center coordinates of n faces, $p_i = (x_i, y_i, t_i)^T$.

(3) Histogram calculation

For a given cuboid $c = (x_c, y_c, t_c, w_c, h_c, l_c)^T$, c is decomposed into $S \times S \times S$ sub-blocks b_i. If the average gradient of each block is $\bar{g}_{bt}$, the histogram h_c of c can be obtained as follows

$$h_c = \sum_{i=1}^{S^3} q_{bt} \tag{6}$$

(4) Calculation of descriptors

Given a sample $s = (x_s, y_s, t_s, \sigma_s, \tau_s)^T$, the spatiotemporal feature points are $(x_s, y_s, t_s)^T$, and the corresponding time dimension and spatial dimension are σ_s, τ_s. Finally, using the formula $r_s = (x_r, y_r, t_r, w_r, h_r, l_r)^T$, select the descriptor ds of the sample s.

Extreme learning machine classifier

Support vector machine (SVM) is widely used in dealing with nonlinear high-dimensional small sample classification problems, and Hussein et al. will also use this method for classification. However, SVM itself has some defects: the algorithm is based on solving quadratic programming and the speed is slow; it is susceptible to the selection of kernel function, penalty factor, and kernel parameter; when dealing with multiclassification problem, the performance is not as good as neural network. To this end, the ELM is selected as the classifier. The main idea of ELM is the

weight parameter between the input layer and the hidden layer, and the offset vector parameters on the hidden layer are determined once. It is not necessary to repeatedly adjust the refresh by iteratively like other gradient-based learning algorithms. It is only necessary to solve a minimum norm least-squares problem and finally classify it into a Moore–Penrose generalized inverse problem for solving a matrix. The algorithm has a few training parameters and is fast.

The ELM basic algorithm is described as follows

1. N: total number of training-samples;
2. $\tilde{N}$: the number of hidden layer units;
3. n, m: the dimensions of the input and output layers;
4. $(x_j, t_j), j = 1, 2, \cdots N$: training-sample, where $x_j = (x_{j1}, x_{j2}, \cdots, x_{jn})^T \in R^n$, and $t_j = (t_{j1}, t_{j2}, \cdots, t_{jn})^T \in R^m$. Combine all output vectors in rows to get the overall output matrix

$$T = \begin{bmatrix} t_1^T \\ t_2^T \\ \vdots \\ t_N^T \end{bmatrix}_{N\times \mathrm{m}} = \begin{bmatrix} t_{11} & \cdots & t_{1m} \\ \vdots & \cdots & \vdots \\ t_{N1} & \cdots & t_{Nm} \end{bmatrix} \tag{7}$$

5. $o_j, j = 1, 2, \cdots, N$: The actual output vector corresponding to the label t_j.
6. $W = (w_{ij})_{\tilde{N}\times n}$ a weight-matrix among the input layer and the hidden layer, wherein the vector $w_i = (w_{i1}, w_{i2}, \cdots, w_{in})^T$ related to the i'th row of W represents a weight-vector connecting the i'th unit and the input unit of the hidden layer.
7. $b = (b_1, b_2, \cdots, b\tilde{N})^T$: offset vector, b_i represents the threshold of the i'th hidden layer unit.
8. $\beta = (\beta_{ij})_{\tilde{N}\times m}$: a weight-matrix among the hidden layer and the output layer, wherein the vector $\beta_i = (\beta_{i1}, \beta_{i2}, \cdots, \beta_{im})^T$ corresponding to the i'th row of β represents a weight-vector connecting the i'th unit and the output-layer unit of the hidden layer. The matrix β can be written in rows as follows

$$\beta = \begin{bmatrix} \beta_1^T \\ \beta_2^T \\ \vdots \\ \beta_{\tilde{N}}^T \end{bmatrix} = \begin{bmatrix} \beta_{11} & \cdots & \beta_{1m} \\ \vdots & \cdots & \vdots \\ \beta_{\tilde{N}1} & \cdots & \beta_{\tilde{N}m} \end{bmatrix} \tag{8}$$

9. $g(x)$: The excitation function. There are five types of excitation functions that are often used, namely the sigmoid function, the sin function, the hardlim function, the tribas function, and the radbas function.

Mathematically, the general model of single hidden layer feedforward neural network (SLFNs) is

$$\sum_{i=1}^{\tilde{N}} (g(w_i \cdot x_j) + b_i)\beta_i = o_j, j = 1, 2, \cdots, N \tag{9}$$

where $w_i \cdot x_j$ delegates the inner product of w_i and x_j.

To make the model of SLFNs close to the above N samples with zero error, it is necessary to

$$\sum_{J=1}^{N} ||o_j - t_j|| = 0 \tag{10}$$

That is, there are W, β and b, so that

$$\sum_{i=1}^{\tilde{N}} (g(w_i \cdot x_j) + b_i)\beta_i = t_j, j = 1, 2, \cdots, N \tag{11}$$

Using the matrix, write the above formula:

$$H\beta = T$$

Wherein, $T \in R^{N\times m}$, $\beta \in R^{\tilde{N}\times m}$. $H = H(W, b) = (H_{ij})_{N\times\tilde{N}}$, here $H_{ij} = g(w_j \cdot x_i + \mathrm{b_j})$, whose i'th row relates to the output vector of the i'th hidden layer unit.

While the number of hidden layer units is not different from the number of samples, that is, $\tilde{N} = N$, and the matrix H is invertible, the equation $H\beta = T$ has a unique solution, that is, the "zero error approaching sample" is described above. In this case, H is a rectangular matrix, and W, b, and β are not necessarily present, so that

$$||H(\hat{W}, \hat{b})\hat{\beta} - T|| = \min_{W,b,\beta} ||H(W, b)\beta - T|| \tag{12}$$

The above equation is identical to minimizing the following cost function

$$E = \sum_{j=1}^{N} \left\| \sum_{i=1}^{\widehat{N}} (g(w_i \cdot x_j) + b_i)\beta_i - t_j \right\|^2 \tag{13}$$

This minimization problem is usually solved using a gradient-based learning method. Remember that $\theta = (W, \beta, b)$ represents the parameters, and then, the corresponding iteration format is

$$\theta_k = \theta_{k-1} - \frac{\partial E(\theta)}{\partial \theta} \tag{14}$$

Usually, in the learning method of SLFNs, the input weight W and the offset vector b of the hidden layer unit are constantly adjusted by iteration. When W and b are fixed, the least squares solution of the linear system $H\beta = T$ can be

$$||H\hat{\beta} - T|| = \min_{\beta} ||H\beta - T|| \tag{15}$$

According to the theorem, Gy is the minimum norm least squares-solution of $Ax = y$ is equivalent to $G = A^+$, then, the least norm least squares-solution of $H\beta = T$ is

$$\beta = H^+ T \tag{16}$$

In the above formula, H^+ delegates the Moore–Penrose generalized inverse matrix of the output matrix H of the hidden layer in the neural network.

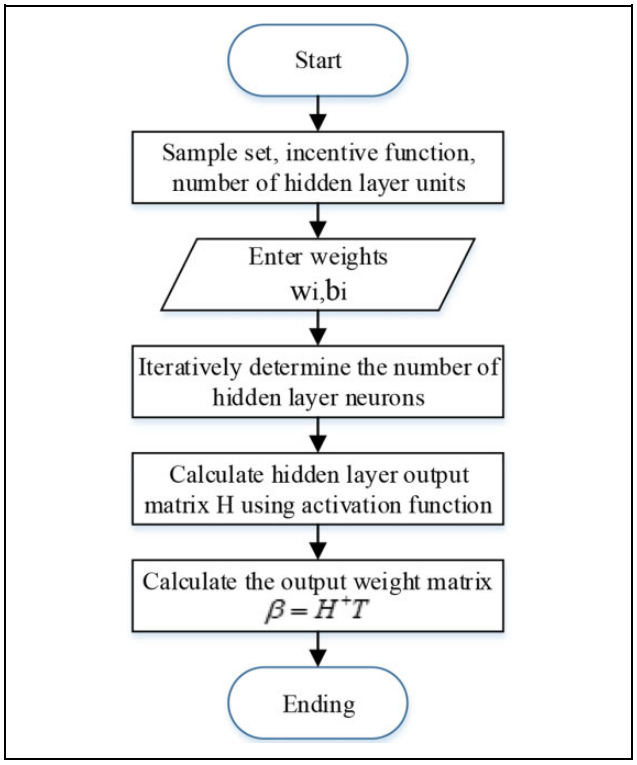

Figure 2. Extreme learning machine algorithm flowchart.

Therefore, the algorithmic flow of the ELM can be the following:

Given the training-sample set $N = \{(\mathrm{x_i}, t_i)|\mathrm{x_i} \in \mathrm{R^n}, \mathrm{t_i} \in \mathrm{R^m}\}_{i=1}^{N}$, the excitation function $g(x)$, the number of hidden layer units $\tilde{N}$.

The first step is to arbitrarily specify the input weights $w_i, b_i, i = 1, 2, \cdots, \tilde{N}$. In the case of small amount of data, it can be selected in an iterative manner to ensure that the recognition effect is similar on the test set and the training set. Too few hidden layer neurons will result in low recognition accuracy. Excessive neurons in the hidden layer will lead to overfitting. The experimental data can be used to see the relevant conclusions, and the connection weights and hidden layer neurons can be determined randomly. The second step is to the activation function to calculate the hidden layer output matrix H.

The third step is to calculate the output weight matrix $\beta = H^{+}T$.

To further illustrate the algorithm process, the algorithm flowchart of the ELM method is shown in Figure 2.

Experiment

Data source

To verify the broad applicability of the method, two data were used for experiments. Part of the data comes from the existing dataset GAMING DATASETS - G3D and UTKinect-Action3D Dataset. The G3D game action dataset includes subject boxing, defense, sports pitching, aiming and shooting, walking, jumping, climbing, crouching, waving, tapping, and applauding. UTKinect-Action3D dataset has 10 types of actions: walk, sit down, stand up, pick up, lift, throw, push, pull, wave, and clap. Another part of the data comes from real-time human motion capture. In the above data, 30% of the data is used for the training set and 70% of the data is used for the test set.

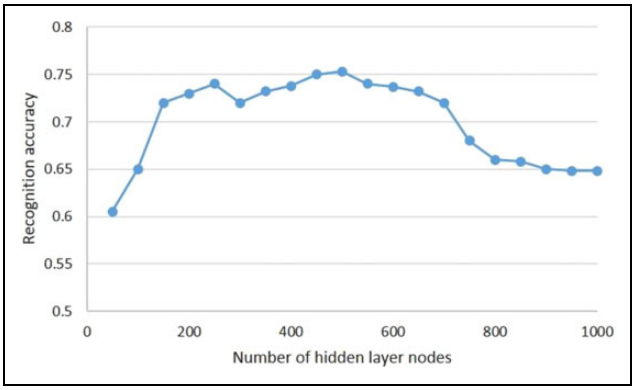

Figure 3. Correspondence between the recognition accuracy rate and the number of hidden layer nodes in GAMING DATASETS-G3D data.

Experimental platform

The human body motion-recognition based on the ELM proposed here is implemented on the PC through the MATLAB language.

Hardware configuration:
CPU: Pentium(R) Dual-Core CPU E5800 @ 3.20 GHz
Memory: 4G
Software configuration:
System: 64 bit win10
Development environment: MATLAB 2014B

Results

Result 1: Correspondence between the correct rate of recognition and the number of hidden layer nodes in different data sets.

The first set of experiments will be carried out in the GAMING DATASETS-G3D and UTKinect-Action3D datasets. First, the optimal parameters of the ELM proposed in this article will be found through experiments, including the influence of the number of nodes in the hidden layer of ELM on the recognition accuracy. To seek out the most appropriate value of the number of hidden layer nodes on different databases, run the ELM method on the above two data sets and select the optimal number of nodes as the number of nodes of the classifier, respectively, and experimental data was recorded every 25 and 50 hidden layer nodes, respectively. The results are shown in Figures 3 and 4.

Figures 3 and 4 show the correspondence between the recognition rate and the number of hidden layer nodes in the two data sets. It can be found that increasing the number of hidden layer nodes within a certain range can improve the recognition accuracy rate. The maximum value of the recognition accuracy is reached on the number of 200 and 500 hidden layer nodes, respectively. However, too many hidden layer nodes are likely to result in overfitting of the training data and increase the computational complexity.

Result 2: Five kinds of motion recognition

In this article, the circumscribing rectangle of the human body contour and contour is extracted, and the aspect values, such as aspect ratio, area, centroid, and offset of the human body contour, are calculated. Dynamically analyze the changes of these features on the time axis, and identify the five types of movements of the human body in the scene: walking, running, jumping, squatting, and lying. The moving targets in different scenes are collected in real time, and the final recognition results are in presented Table 1.

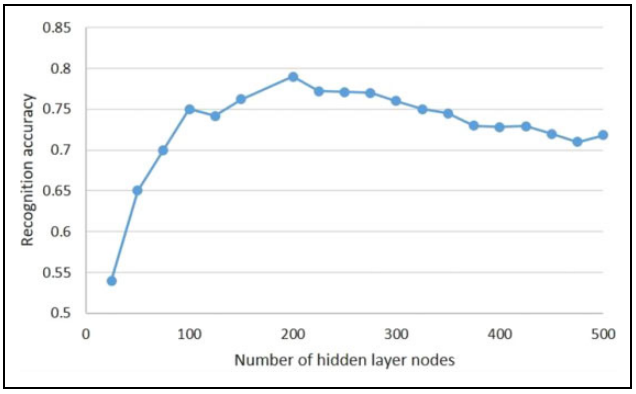

Figure 4. Correspondence between the recognition accuracy rate and the number of hidden layer nodes in UTKinect-Action3D data.

Table 1. Five kinds of motion recognition results.

Actual behavior result	Walk	Run	Jump	Lie	Squat
Walk	94.2	7.3	5.8	1.3	0
Run	5.9	93.1	2.2	0	0
Jump	0	0	92.0	0	0
Lie	0	0	0	99.3	0
Squat	0	0	0	0	100

As can be seen from Table 1, due to the difference in size, wearing, and different speeds of the human body, some recognition errors may occur: if a person walks too fast, it will be mistaken for a person running, and vice versa. Then, it is mistaken for walking; when the person jumps, the offset of the centroid in the vertical direction and the coordinates of the circumscribed rectangle have obvious changes, but at the moment when the jump is completed, it is misjudged as walking or running; the recognition process under the arm is similar to jumping; lying is the most special state of motion in the horizontal aspect ratio.

The correct recognition rate of the five actions in Table 1 is quite high, but there are also some cases of recognition errors. This is because this article mainly uses the shape and motion characteristics of the human body and regards the human motion process as a continuous combination of different postures. The similar posture will cause a certain misunderstanding rate. Nevertheless, through a large number of experimental tests, the proposed method achieves satisfactory results, not only can accurately detect human moving targets in dynamic video sequences but also identify some actions based on human body shape and motion information.

Result 3: Effect of noise on recognition rate

The experimental data in this article are collected in an environment close to the ideal environment, that is, there is always only one experimenter in the communication environment, and there are no other interfering objects in the room. However, there must be a lot of noise in the real use

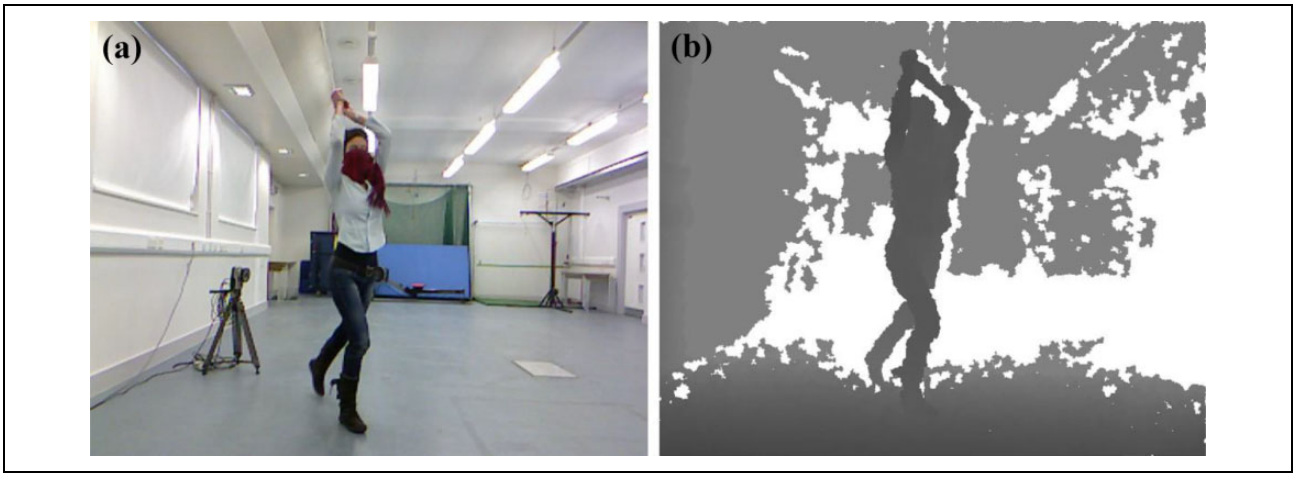

Figure 5. (a, b) Human motion data collection.

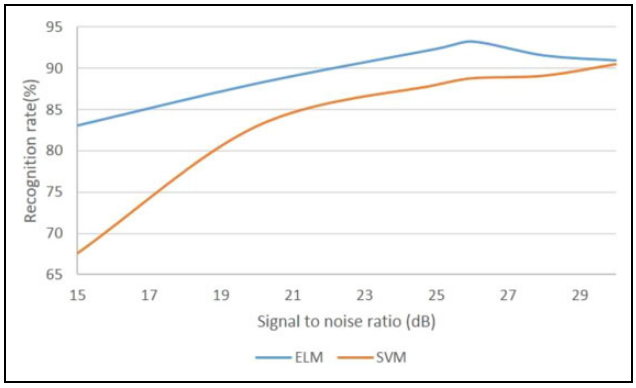

Figure 6. Recognition rate under different SNR. SNR: signal-to-noise ratio.

scene, or there are multiple athletes. The data collected in this article and the human target recognition are shown in Figure 5. At present, the widely used classifier in the target recognition algorithm is SVM. SVM has outstanding performance in solving small sample problems and high dimensional indivisible problems. In this section, by adding white-noise to the collected data, the difference between the recognition rate and the SVM-based classifier is compared under different SNR conditions. The result is shown in Figure 6.

In the above analysis process, we assume that the original data set is a pure gesture signal, so only the white noise can be added to obtain the change of the recognition rate under different SNR. In this experiment, the recognition effect of SVM and ELM is analyzed from the condition of 15–30 dB different SNR. From Figure 6, the human motion is affected by noise. With the increase of SNR, the recognition rate of the two methods changes in different degrees. When the SNR is less than 26 dB, the recognition rate of SVM method increases rapidly and then slows down, while the recognition rate of ELM method increases steadily. However, in the statistical range, the recognition rate of the proposed method is always higher than that of SVM. The ELM-based recognition method in this article is much better than the SVM-based recognition method and has great advantages.

Conclusions

Human motion recognition has broad application prospects and theoretical research value, and has received more and more attention in recent years. In the research of this article, the author first carries out data preprocessing, detects the human motion area to determine the location of the target, and prepares for the next feature extraction. Then, the interframe difference method is used to obtain the area, where each frame of the moving target is located, and extract effective and representative action characteristics; Finally, a classifier with good performance is established using the ELM method to achieve the recognition of human motion. The simulation results demonstrate that the application of the ELM to human motion recognition has achieved good results and has a good reference value in the field of human motion recognition.

Although there are many studies on human motion recognition technology, human behavior recognition is still a challenging open topic due to the complex and versatile human motion. Firstly, choosing the appropriate motion representation is still a problem to be studied. Secondly, in real life, the scene is complex, the lighting conditions are complex, and occlusion will affect the motion recognition effect. As the category of human motion increases, similar actions of each category interfere with each other, which reduces the resolving power of the features. Therefore, the practicality of human motion recognition is also a difficult problem to be solved; again, due to the constraints of the existing database, the human motion recognition based on the ELM has a broad research prospect.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

ORCID iD

Feiping Liu https://orcid.org/0000-0003-1480-1645

References

1. Poppe R. Vision-based human motion analysis: an overview. *Comput Vis Image Understand* 2007; 108(1-2): 4–18.
2. Ji X and Liu H. Advances in view-invariant human motion analysis: a review. *IEEE Trans Syst Man Cybern Part C* 2009; 40(1): 13–24.
3. Shengwei C, Bingqian H, Siyu C, et al. Human motion recognition based on time and space points of interest. *J Chengdu Univ Inform Technol* 2018; (2).
4. Johansson G. Rigidity, stability, and motion in perceptual space: a discussion of some mathematical principles operative in the perception of relative motion and of their possible function as determinants of a static perceptual space. *Acta Psychol* 1958; 14: 359–370.
5. O'Rourke J and Badler NI. Model-based image analysis of human motion using constraint propagation. *IEEE Trans Pattern Anal Mach Intell* 2013; PAMI-2(6): 522–536.
6. Nagel HH. From image sequences towards conceptual descriptions. *Image Vis Comput* 1988; 6(2): 59–74.
7. Yamato J, Ohya J, and Ishii K. Recognizing human action in time-sequential images using hidden Markov model. *Trans Instit Electron Inform Commun Eng* 1993; 76(9): 379–385.
8. Polana R and Nelson R.Low level recognition of human motion (or how to get your man without finding his body

parts). In: *IEEE workshop on motion of non-rigid & articulated objects*. IEEE, Austin, TX, USA, 11–12 November 1994, pp. 77–82.
9. Davis J and Bobick A. The representation and recognition of human movement using temporal templates. In: *IEEE computer society conference on computer vision & pattern recognition*, San Juan, Puerto Rico, USA, 17–19 June 1997, pp. 928–934.
10. Gritai A, Sheikh Y, and Shah M. On the use of anthropometry in the invariant analysis of human actions. In: *Proceedings of the 17th IEEE computer society international conference on pattern recognition*, Cambridge, UK, 26–26 August 2004, pp. 923–926.
11. Collins RT, Lipton AJ, and Kanade T. A system for video surveillance and monitoring: VSAM final report. Robotic Institute Carnegie Mellon University: Technical Report: CMU-RI-TR, 2000, pp. 1–68.
12. Matsuyama T. Cooperative Distributed Vision: dynamic integration of visual perception, action, and communication. In: *Proceedings of image understanding workshop*. 1999, pp. 75–88.
13. Renard M. Annotated digital video for intelligent surveillance and optimitized retrieval: advisor final evaluation report. ADVISOR-DOC-039, 2003, pp. 1–95.
14. ISCAPS: integrated surveillance of crowded areas for public security, http://www.iscaps.reading.ac.uk/.
15. Wang X, Ma X, and Grimson E. Unsupervised activity perception by hierarchical Bayesian models. In: *IEEE conference on computer vision & pattern recognition*. Minneapolis, MN, USA, 17–22 June 2007, pp. 1–8.
16. Urtasun R and Darrell T. Sparse probabilistic regression for activity-independent human pose inference. In: *IEEE conference on computer vision & pattern recognition*. IEEE, Anchorage, AK, USA, 23–28 June 2008, pp. 1–8.
17. Yilmaz A and Shah M. Actions sketch: a novel action representation. In: *IEEE computer society conference on computer vision and pattern recognition*, San Diego, CA, USA, 20–25 June 2005, pp. 984–989.
18. Niebles JC and Li FF. A hierarchical model of shape and appearance for human action classification. In: *IEEE conference on computer vision and pattern recognition*, Minneapolis, MN, USA, 17–22 June 2007, pp. 1–8.
19. Natarajan P and Nevatia R. View and scale invariant action recognition using Multiview shape-flow models. In: *IEEE conference on computer vision & pattern recognition*. IEEE, Anchorage, AK, USA, 23–28 June 2008, pp. 1–8.
20. Dalal N and Triggs B. Histograms of oriented gradients for human detection. In: *IEEE computer society conference on computer vision and pattern recognition*, San Diego, CA, USA, 20–25 June 2005, pp. 886–893.
21. Zhaoqi W, Yongdong Z, and Shihong X. Three-dimensional human motion simulation and video analysis system for sports training. *Comput Res Develop* 2005; 42(2): 344–352.
22. Zhang X, Liu Y, and Huang TS. Motion analysis of articulated objects from monocular images. *IEEE Trans Pattern Ana Mach Intell* 2006; 28(4): 625–636.
23. Ramanathan V, Liang P, and Li FF. Video event understanding using natural language descriptions. In: *IEEE International Conference on Computer Vision*, Sydney, NSW, Australia, 1–8 December 2013, pp. 905–912.
24. Liang W, Weiming H, and Tieniu T. Gesture-based identification. *Chin J Comput* 2003; 26(3): 353–360.
25. A new method for feature extraction of illumination invariants and its application in target recognition. *Chin J Electron* 2018; 46(4): 895–902.
26. Huang GB, Zhu QY, and Siew CK. Extreme learning machine: theory and applications. *Neurocomputing* 2006; 70(1–3): 489–501.
27. Ding S, Zhao H, Zhang Y, et al. Extreme learning machine: algorithm, theory and applications. *Artif Intell Rev* 2013; 44(1): 103–115.
28. Zuoren L, Jiayu W, Yutong A, et al. Application of extreme learning machine in image segmentation. *Comput Knowl Technol* 2016; 12(3): 207–209.
29. Songlin K, Le L, Chuchu L, et al. Application of multi-layer extreme learning machine in intrusion detection. *J Comput Appl* 2015.
30. Xiangming G and Jianmin L. Application of extreme learning machine in intelligent diagnosis of motor fault. *Meas Control Technol* 2015; 34(8): 12–15.