Vol. 40(2), pp. 45-57, Dec. 2021
ISSN 1821-536X (print)
ISSN 2619-8789 (electronic)

Tanzania Journal of Engineering and Technology

Full Length Research Paper

Real-time Complex Hand Gestures Recognition Based on Multi-Dimensional Features

Isack Bulugu

University of Dar es Salaam, College of Information and Communication Technologies
Department of Electronics and Telecommunications, P.O. Box 33335
bulugu@udsm.ac.tz

ABSTRACT

Gesture recognition is broadly utilized within the field of sensing. There are basically three gesture recognition methods based on computer vision, depth sensor and motion sensor. Motion sensor-based gesture recognition has few input data, fast speed, and direct access to three-dimensional information of the hand. The advantages of traditional motion sensor-based gesture recognition have gradually become a current research hotspot. The essence of traditional motion sensor-based gesture recognition is a pattern recognition problem, and its accuracy depends heavily on the feature dataset extracted from prior experience. Unlike traditional pattern recognition methods, deep learning can be used to a large extent, reducing the workload of artificial heuristic extraction of features. In order to solve the problems of traditional pattern recognition, this paper proposes a real-time recognition method of multi-feature gestures based on a long short-term memory network (LSTM), which is verified by sufficient experiments. The method first defines a gesture library of five (5) basic gestures and seven (7) complex gestures. Based on the kinematic characteristics of the hand posture, the angle features and displacement features are further extracted, and then short-time Fourier transform (SFTF) is used. The frequency domain features of sensor data are extracted, and the three features are input into the deep neural network LSTM to train, classify and recognize the collected gestures. At the same time, to verify the effectiveness of the proposed method, a self-designed handheld experience stick is collected. The gesture data of six (6) volunteers is used as an experimental data set. The collected experimental results show that the proposed recognition method has a recognition accuracy of 93.50% for basic and complex gestures. Compared with other methods, the recognition accuracy has increased by nearly 2%.

Key words: *Human-computer interaction; gesture recognition; inertial sensor; motion capture.*

INTRODUCTION

Gestures provide people with a natural and intuitive way of communication. Gesture recognition refers to the process of categorizing actions such as the degree of bending of the fingers, the angle of the palm flip, and the length of the arm to understand the meaning of an expression. It can greatly simplify the human-computer interaction process (Chu Y, 2020) compared with traditional input methods such as mouse and keyboard. People can omit the intermediate medium of human-computer interaction and control the machine more intuitively and conveniently. For example, Kundu (2018) introduced a gesture-based intelligent wheelchair motion control system that uses five (5) gestures to control the wheelchair to turn left, right, forward, back and stand-still. Also, Zou Z (2020) uses MPU6050 six-axis attitude sensor-combined with Arduino smart car via radio frequency to design a gesture recognition system that can realize the function of controlling the direction of the smart car through gesture changes. Moreover, Zhou (2020) developed a virtual piano through a software platform, supplemented by gesture recognition to achieve no physical performance. In addition, gesture recognition can also be applied in the fields of deaf-mute sign language recognition, game manipulation, and medical rehabilitation (Wang J S, 2012) (Zhang X, 2011) (Lu Z, 2017) (Liu Y H, 2014) (Zhao N, 2020) (Oliver A) (Chen L, 2019). For example, in clinical surgery, the use of Kinect-based gesture recognition allows doctors to use simple gestures to check CT scan results, nuclear magnetic field, etc., in a sterile environment. Resonance images and other medical images are scaled, moved, etc. (Zhu T, 2013).

According to the different monitoring methods, gesture recognition is mainly divided into three methods based on computer vision, depth sensors and motion sensors (Chen L C, 2013).

.

METHODS AND MATERIALS

Related Works

The traditional motion sensor-based gesture recognition method is essentially a pattern recognition. In some specific knowledge fields (such as disease prediction) or other controllable scenarios, traditional pattern recognition methods (such as decision trees, support vector machines (Miao Y W, 2020), Naive Bayes and Hidden Markov Models (Oscar D L, 2013), etc.) have made great progress. However, in gesture recognition tasks, these methods rely heavily on features extracted by artificial heuristics based on prior experience (Y, 2013) (Nafea O, 2021). Leading to their unsupervised and incremental tasks, the performance is reduced. Unlike traditional pattern recognition methods, deep learning can greatly reduce the workload of artificial heuristic extraction of features, which can be learned by training end-to-end neural networks with more high-level, meaningful features. In addition, the deep network structure is more suitable for incremental learning.

Deep learning is an ideal method for gesture recognition, and it has been extensively explored in existing work (Alsheikh M A, 2015) (Lane N D, 2015) (Thomas P, 2011). (Vepakomm A P, 2015). First, features are extracted manually from sensor data based on prior experience and then input these features into deep neural networks. Similarly, Walse et al. (Walse K H, 2016) used the principal component analysis method to extract features, and then input the features into a deep neural network for gesture recognition. In these works, the deep neural network is only used as a classification model after manual feature extraction, so it is not obtained. A modified network structure is quite simple, and deep feature training cannot be carried out. The author in (Hammerlar N Y, 2016) used a deep neural network with five (5) hidden layers for automatic feature learning and classification to improve recognition performance. This shows that when the gesture recognition data is multi-dimensional, and the activities are more complex, because the hidden layer has stronger representation ability, more hidden layers can help the model to train better. However, the performance of these methods heavily depends on the training set, the algorithm stability is poor. Literatures Almaslukh B (2017)and Wang A (2016) apply stacked auto-encoders to gesture recognition, first using the greedy algorithm, pre-processing layer by layer, and then fine-tuning the parameters.

In contrast, Li Y (2014) uses sparse auto-encoders to improve the performance of gesture recognition by adding KL divergence and noise parameters to the cost function. The advantages of stacked auto-encoders are that they can perform unsupervised feature learning on sensor data and can be used as an effective tool for feature extraction. However, stacked auto-encoders rely too much on the input layer, hidden layer, output layer and activation function, and it is difficult to find the optimal solution.

Therefore, to solve the above problems, this paper proposes a real-time recognition method for multi-feature gestures based on a long short-term memory network (LSTM), which models the contextual spatial relationship of hand movements and detects gestures in real-time. In order to implement the proposed method, this paper proposes a handheld somatosensory stick, which monitors hand movement information through a three-axis accelerometer, a three-axis gyroscope, and a three-axis magnetometer. In order to eliminate the recognition errors caused by the different speeds of gestures performed by different users, the hands are first extracted, the angular feature, displacement feature, and frequency domain feature of the sensor data. The sliding window is then used to segment the time series sequence representing the motion feature; then the LSTM is used to perform context-related iterative training on the feature sequence to obtain the final gesture recognition model.

The main contributions of this paper are as follows:

(1) The design of a handheld somatosensory stick can effectively collect the user's gesture data and provide good data support for subsequent model training.

(2) Multi-dimensional extraction of user gesture motion features, extracting the angle features of hand motion, displacement features, and frequency domain features of sensor data, multi-dimensional features help improve the accuracy of gesture recognition.

The rest of the paper is as follows: Section 3 of this article mainly introduces the basic structure of the sensor system. Meanwhile, Section 4 discusses the method proposed in this article in detail while Section 5 mainly focus on the design and effect of the verification experiment. The conclusion and recommendation are drawn in the final section.

An Overview of a Sensing System

In order to verify the performance of the proposed method, this article independently designed a handheld somatosensory stick, which can obtain experimental training data more effectively and provides a wealth of verification data for subsequent experiments. The handheld somatosensory stick is composed of a micro-control unit (Arduino Nano). The axis motion sensor (MPU9250) and the wireless communication module (HC-08) are composed of a virtual serial port and the host computer for information transmission, as shown in Figure 1 below.

Arduino Nano is a microcontroller based on ATmega32. Its size is similar to the size of a coin, and is suitable for wearable devices. MPU9250 is a nine-axis attitude sensor that integrates a three-axis accelerometer, a three-axis gyroscope and a three-axis magnetometer, which reduces the size and power consumption of the module through integration. The motion sensor is connected via a bus. The microcontroller unit pre-processes the raw sensor data, and the sensor data reading rate in the experiment is 40 Hz.

The wireless communication module is a Bluetooth module based on the Bluetooth SpecificationV4.0 protocol. It has the characteristics of low power consumption and fast transmission rate. The wireless working frequency band is 2.4GHzISM, the maximum transmission power is 4dBm, and the receiving sensitivity is -93dBm.

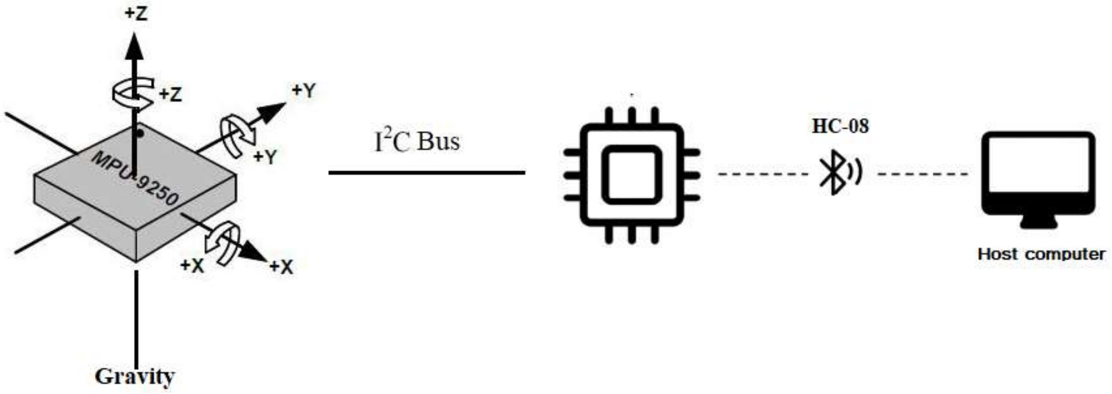

Figure 1: Schematic of hand-held somatosensory stick

Real-time Recognition of Multi-feature Gestures based on LSTM

The multi-feature gesture real-time recognition system based on LSTM is composed of the following modules; signal acquisition and data pre-processing, feature extraction, data segmentation, and LSTM classifier. The system defines five (5) basic gestures (see Figure 2 below) and seven (7) complex gestures (see Figure 3 below). Since the basic gesture recognition process is part of the complex gesture recognition process, this article will focus on the entire process of complex gesture recognition. Hand motion information is measured by a nine-axis motion sensor and then passed through a median filter and a low-pass filter. The processed data extracts the angle feature and the displacement feature. Then the sensor data is extracted through the short-time Fourier transform (SFTF) to extract the frequency domain feature. After the feature extraction, the time series data is segmented using the sliding window. Finally, the LSTM, the classifier, recognizes the predefined gestures and obtains the gesture label. The implementation steps of the gesture recognition method proposed in this paper are shown in Figure 4 below.

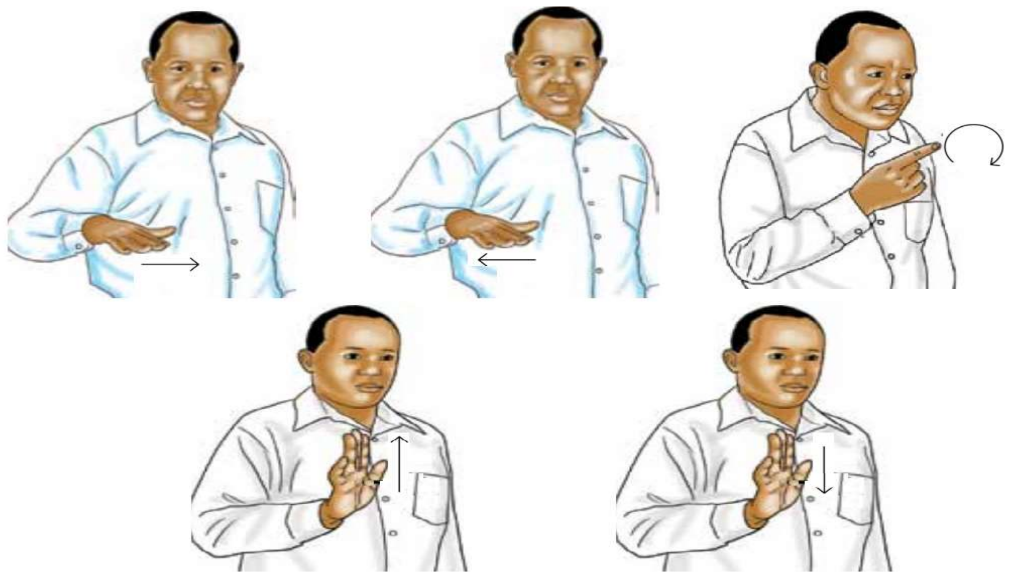

Figure 2: Simple hand gestures

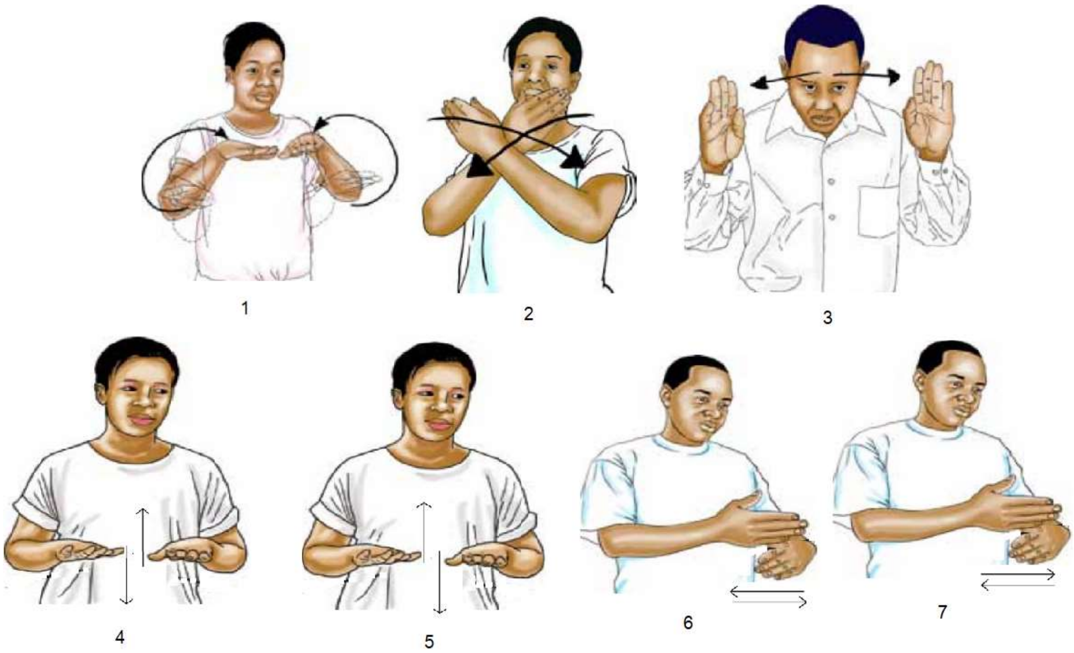

Figure 3: Complex hand gestures

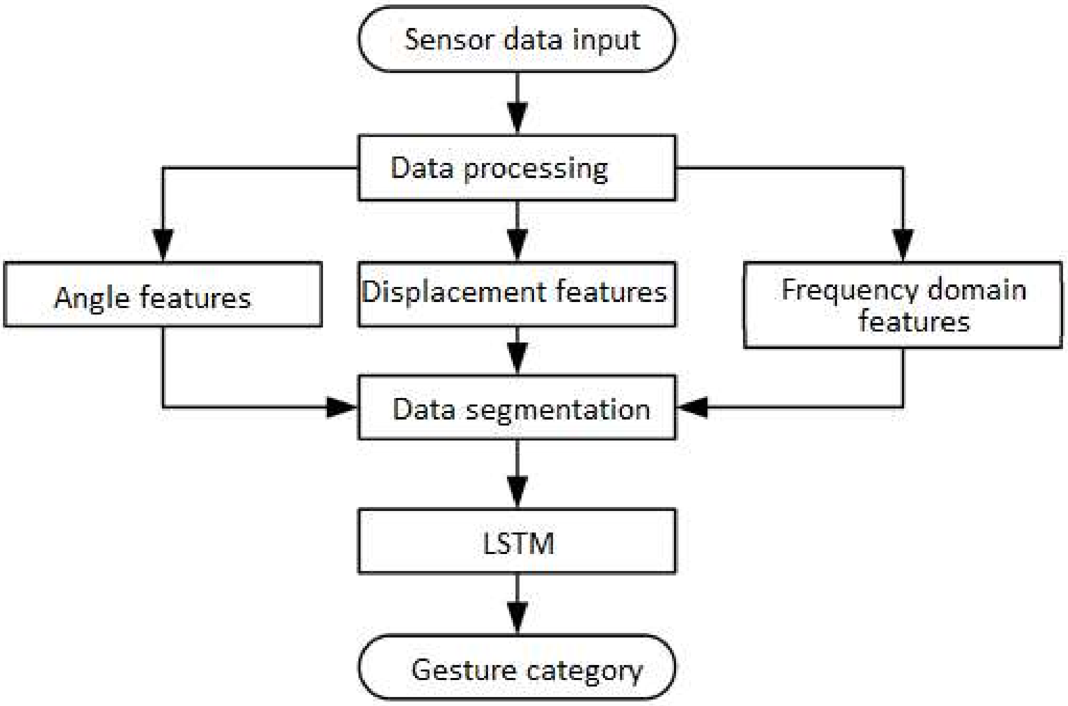

Figure 4: Flow chart of real-time hand gesture recognition

Data Collection

The experimental data in this article is mainly obtained by the handheld somatosensory stick designed in Section 3. At the same time, in order to collect reliable gesture data, the experimental subjects followed the following guidelines during the data collection stage:

- During the entire data collection process, the handheld somatosensory bar is placed horizontally (the Z axis of the sensor chip is parallel to the direction of gravity).

- The time interval between every two gestures is not less than 0.2s, so that the data segmentation program can segment each gesture in sequence.
- Perform pre-defined gestures, as shown in Figure 2 and Figure 3.

The original signal of hand movement is generated by a three-axis accelerometer, a three-axis gyroscope and a three-axis magnetometer, and is collected by a single-chip microcomputer. In order to reduce the impact of accidental hand movements, this article adopts a simple calibration method. First, the hands slowly carries out a “8” shape movement, the offset is obtained by calculating the average of the maximum and minimum values, and then the offset obtained is subtracted. Similarly, slowly rotate along the Z axis to eliminate the offset of the hand on the horizontal plane to calibrate the accelerometer. Finally, record the maximum and minimum values of the gyroscope when it is stationary to calibrate the gyroscope's bias error.

Data preprocessing

After the sensor data is calibrated, it is also necessary to use a median filter to smooth the data and reduce data abnormalities caused by hand shaking. The processing expression of the median filter is:

$$R_p(n) = \frac{1}{2M+1}\sum_{m=-M}^{M} R(n+m) \qquad (1)$$

where R is the original sensor signal before processing and R_p is the sensor data after median filtering. In this paper, M is set to 5, that is, the filter window size is 11.

According to the characteristics of the motion sensor signal, the data collected by the accelerometer and the magnetometer respectively have the earth's gravitational field component and the geomagnetic field component, and should be fixed when the direction is unchanged. Therefore, the high-frequency signal contributes more to the action description. In order to further improve the availability of data, this article uses low-pass filters for accelerometers and magnetometers to reduce data interference caused by high-frequency signals and reduce data noise.

When designing a filter, the cut-off frequency is an important indicator that needs to be considered. For hand motion, the maximum autonomous motion frequency should be less than 6 Hz (Xiong Y, 2006). In order to verify this information, a hand-held somatosensory wand is used multiple times to perform predefined gestures. The changes of the original data during the autonomous movement of the hand are measured. In the end, according to the actual test results, the cut-off frequency of the filter is set to 20 Hz, and the order is 2. Figure 5 below shows the result of preprocessing of the X-axis signal of the gyroscope.

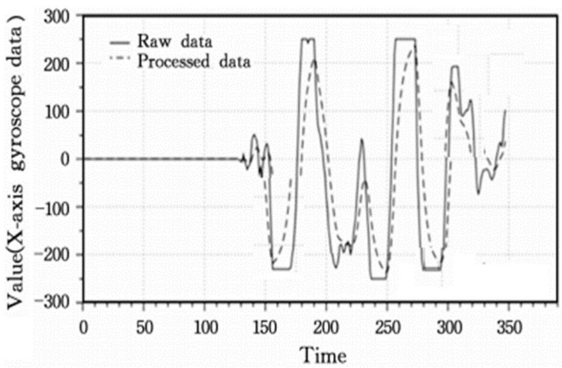

Figure 5: Results after data pre-processing

Multi-Dimensional Feature Extraction

Hand movement will produce complex motion trajectories, and this paper extracts three features; angle feature, displacement feature, and frequency domain feature multi-dimensionally for gesture recognition. Hand movement can be described by angle feature and displacement feature, and the total angle is used. Angle change represents the angle feature to avoid recognition errors due to different users’ different speeds of gestures. The total angle change $\Delta\theta_n$ is obtained by integrating the sequence of changes in angular velocity ω in n data points.

However, one feature value is not enough to uniquely describe the entire movement; therefore, it is necessary to construct a feature vector of an angle. The feature vector is constructed from the total angle change in different periods. The expression of the feature vector is given by (2).

$$\Delta\theta_i = \sqrt{\left(\left(\sum_{j=1}^{i}\omega_{jx}\Delta t\right)^2 + \left(\sum_{j=1}^{i}\omega_{jy}\Delta t\right)^2 + \left(\sum_{j=1}^{i}\omega_{jz}\Delta t\right)^2\right)} \qquad (2)$$

where ω_{jx} is the x-axis data of the j-th group of gyroscopes; ω_{jy} is the y-axis data of the j-th group of gyroscopes; ω_{jz} is the z-axis data of the j-th group of gyroscopes; Δt is the interval time between adjacent sampling points; $\Delta\theta_1$ is the total angle change during the first Δt; $\Delta\theta_1$ is the total angle change during the first two Δt periods and $\Delta\theta_n$ is the total angle change during the first $n\Delta t$ periods. The displacement characteristic is expressed by the displacement change and is obtained by integrating the linear acceleration. The three-axis accelerometer measures the linear acceleration due to hand movement. This article double-integrates the data collected by the accelerometer to obtain the displacement characteristic $\Delta\lambda$. The expression of the displacement characteristic $\Delta\lambda_n$ at the nth time is given by equation (3). Where α_{jx}, α_{jy}, and α_{jz} respectively represent the data of the accelerometer on the x-axis, y-axis, and z-axis at the j-th time.

The angle and displacement features are both features in the time domain. This article additionally uses Short-Time Fourier Transform (STFT) to process sensor data. By transforming the time domain features, the extraction is not obvious in the time domain and it is easy to be ignored.

The specific formula of the Short-Time Fourier Transform is as follows:

$$F(\omega,t) = \int_R f(t)g(t-\tau)e^{it\omega}dt \qquad (4)$$

where, $f(t)$ is the input sensor data, and $g(t)$ is the window function.

By short-time Fourier transform of $f(t)$, a local slice of $f(t)$ near time $t=\tau$ is obtained. can $F(\omega, t)$ roughly reflect the relative proportion of the signal component of the frequency of ω when the input data $f(t)$ is at time t, so the information of the sensor data in the frequency domain and the time domain can be obtained through this transformation.

Data Segmentation

The data segment that records the gesture action process in the motion data sequence is called the active segment. Gesture recognition needs to automatically determine the start and end points of the active segment from the continuous input signal stream. Due to the context and space relationship of body motion and the segmentation of gesture data, the process is more difficult.

This article uses a distance-based sliding window method to segment the time series data stream. As shown in Figure 6 below, the distance between adjacent sample points in the data stream when the user is moving is much greater than the distance when the user is stationary, and the threshold can be used very much.

Quickly finding the starting point of the active segment, and then using the sliding window method to intercept the data stream, you can get the active segment for identification. It should be noted that this article uses the Euclidean distance to measure the distance between adjacent sample points, and at the same time according to the experiment, the average time for the user to perform gestures set the window size for intercepting the data stream to 60.

$$\Delta\lambda_n = \left\{ \left(\Delta\lambda_{nx}, \Delta\lambda_{ny}, \Delta\lambda_{nz} \right) \middle| \left(\Delta\lambda_{nx} = \sum_{j=1}^{n} \alpha_{jx}\Delta t, \Delta\lambda_{ny} = \sum_{j=1}^{n} \alpha_{jy}\Delta t, \Delta\lambda_{nz} = \sum_{j=1}^{n} \alpha_{jz}\Delta t \right) \right\} \quad (3)$$

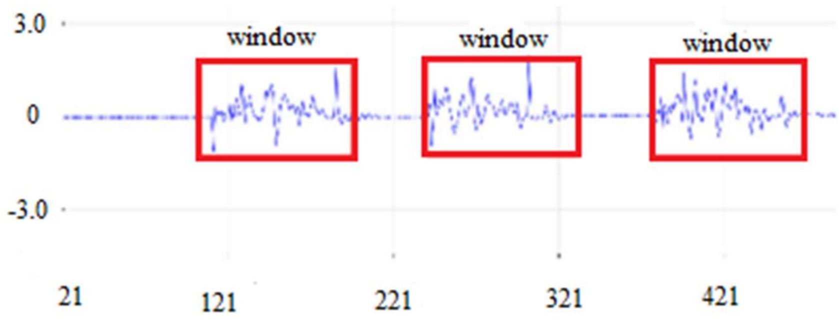

Figure 6: Data stream segmentation.

Classifier Construction

Angle features and displacement features can represent hand movement and help improve gesture recognition performance. However, it can only express the spatial information of hand movement. Gesture movement has continuity, and adjacent gestures are interrelated. Therefore, LSTM has the ability to learn long-term dependence on the information. Since gesture movement is a collection of continuous actions, the actions of each frame will have a certain correlation with the actions of the previous frame. Therefore, the use of LSTM to recognize gestures has natural advantages.

LSTM uses memory cells to store and output information. LSTM has a special structure called a "gate" to remove or add information to the cell, and update the cell state of the LSTM according to the activation of the gate. The input provided to the LSTM will be three different kinds of gates, namely, forget gate, input gate and input gate. The following is the specific formula:

$$\boldsymbol{i}_t = \sigma_i\left(\boldsymbol{W}_{ai}\alpha_t + \boldsymbol{W}_{hi}\boldsymbol{h}_t + \boldsymbol{W}_{ci}\boldsymbol{c}_{t-1} + \boldsymbol{b}_i\right) \quad (5)$$

$$\boldsymbol{f}_t = \sigma_f\left(\boldsymbol{W}_{af}\alpha_t + \boldsymbol{W}_{hf}\boldsymbol{h}_t + \boldsymbol{W}_{cf}\boldsymbol{c}_{t-1} + \boldsymbol{b}_f\right) \quad (6)$$

$$\boldsymbol{c}_t = \boldsymbol{f}_t\boldsymbol{c}_{t-1} + \boldsymbol{i}_t\sigma_c\left(\boldsymbol{W}_{ac}\alpha_t + \boldsymbol{W}_{hc}\boldsymbol{h}_t + \boldsymbol{W}_{ci}\boldsymbol{h}_{t-1} + \boldsymbol{b}_c\right) \quad (7)$$

$$\boldsymbol{o}_t = \sigma_o\left(\boldsymbol{W}_{ao}\alpha_t + \boldsymbol{W}_{ho}\boldsymbol{h}_{t-1} + \boldsymbol{W}_{co}\boldsymbol{c}_t + \boldsymbol{b}_o\right) \quad (8)$$

$$\boldsymbol{h}_t = \boldsymbol{o}_i\sigma_i\left(\boldsymbol{c}_t\right) \quad (9)$$

where, i, f, o and c are the activation vectors of the input gate, forget gate, output gate, and cell, respectively. The size of all these vectors is the same as the vector $\boldsymbol{h}$ used to define the hidden value, and σ represents a nonlinear function. α_t represents the input to the cell state at time t: $\boldsymbol{W}_{ai}$, $\boldsymbol{W}_{hi}$, $\boldsymbol{W}_{ci}$, $\boldsymbol{W}_{af}$, $\boldsymbol{W}_{hf}$, $\boldsymbol{W}_{cf}$, $\boldsymbol{W}_{ac}$, $\boldsymbol{W}_{hc}$, $\boldsymbol{W}_{ao}$, $\boldsymbol{W}_{ho}$, and $\boldsymbol{W}_{co}$, are the weight matrices, and the subscript represents the relationship from a certain door to a certain door. For example, $\boldsymbol{W}_{ai}$ represents the matrix from the input gate to the output gate, and $\boldsymbol{W}_{hi}$ represents the hidden input gate matrix whereas b_i, b_f, b_c and b_o are bias vectors.

RESULTS AND DISCUSSION

Experimental Data

The experimental data were collected from six (6) volunteers (3 males and 3 females) holding somatosensory sticks to perform predefined gestures. In order to fully evaluate the performance of the real-time gesture recognition method proposed in this article, each volunteer was asked to perform each gesture repeatedly. The execution speeds of the 20 gestures were slower (the execution time of each gesture does not exceed 2s); normal (the execution time of each gesture does not exceed 1.5s);

relatively slow and fast (the execution time of each gesture does not exceed 1s). The camera is used to synchronously record the user's hand movement during the data collection process; then use the visualization tool to manually segment and label the collected data by comparing the sensor data with the video data.

Experimental Environment and Parameters

The model in this article is built on the Ubuntu system (version 16.04.3 LTS) and implemented in Python (Tensor flow 1.14.0, Sklearn 0.23.1, Pandas 1.0.5, Numpy 1.18.5, Matplotlib 3.2.2 and Keras 2.2.5).

In the experiment, a network structure composed of 2 LSTM layers and 1 SoftMax layer is designed. First, the input layer is passed through the tanh function, and 72 features are extracted from the data stream with the input window length of 11 as an input. Then L2Gnorm and Adam are set. The optimizer uses gradient descent to train the model parameters. Finally, the SoftMax function outputs 12 predefined gesture class labels.

In the training phase, the data collected from 6 volunteers is input into the LSTM, and the window size of the time series is set to 60. When the model is trained, the parameters of the model are updated at a learning rate of 0.0025. In addition, the data is divided into batches. In processing, after each set of data is input, the accuracy of the current training model applied to the test set is tracked, and after all the data is trained, results from the final model are obtained.

Experimental Evaluation Index

The essence of gesture recognition is a classification problem, and the classifier confusion matrix is listed in Table 1 below.

Table 1: Classifier confusion matrix

	Predicted Positive Samples	Predicted Negative Samples
Actual Positive Samples	TP	FN
Actual Negative Samples	FP	TN

This article adopts the evaluation criteria such as accuracy rate (ACC), precision rate (P), recall rate (R) and F1 value. The ACC shows the ability of the classification model to correctly judge the sample type, which represents the classification model's ability to perform classification and recognition, and is calculated as shown by (10):

$$ACC = \frac{TP + TN}{TP + TN + FN} \tag{10}$$

The precision rate (P) also represents the proportion of samples that are correctly judged as positive in all samples judged as positive. Thus P is calculated using the formula indicated by (11) as:

$$P = \frac{TP}{TP + FN} \tag{11}$$

Recall rate (R) which also represents the proportion of samples that are correctly judged as positive in all positive samples. The calculation formula is as follows:

$$R = \frac{TP}{TP + FN} \tag{12}$$

Experimental Evaluation and Results

Three experiments are designed to evaluate the real-time gesture recognition method proposed in this paper. The first experiment is to evaluate the recognition performance of the algorithm under the condition of user dependence; the second experiment requires two testers to be recruited to verify that the algorithm is in Recognition performance in the case of user independence; the third experiment is to evaluate the real-time performance of the algorithm. User dependence means that the test system and training model data are the same people, and user independence means that the test system and training model data come from different people. The main

distinction is to verify the robustness of the system and the generalization ability of the model.
In the experiment of the recognition result under the condition of user dependence, the ten-fold crossover method is used to evaluate the recognition of each person who wishes. Three evaluation indicators (accuracy rate, precision rate and recall rate) are used, and the method will be proposed contrast with support vector machine, K- nearest neighbour method and fully connected neural network (including two fully connected hidden layers, and using the softmax function to output predefined gesture class labels). Figure 7 below shows the overall performance comparison of 12 gestures, simple gestures and complex gestures. The results show that the accuracy of the method proposed in this paper is 93.50% in the case of user dependence, and the overall performance is higher than the other three methods. This is mainly because the LSTM-based recognition method understands the context of the data stream by selecting and leaving historical information, and has a stronger recognition performance for time series.

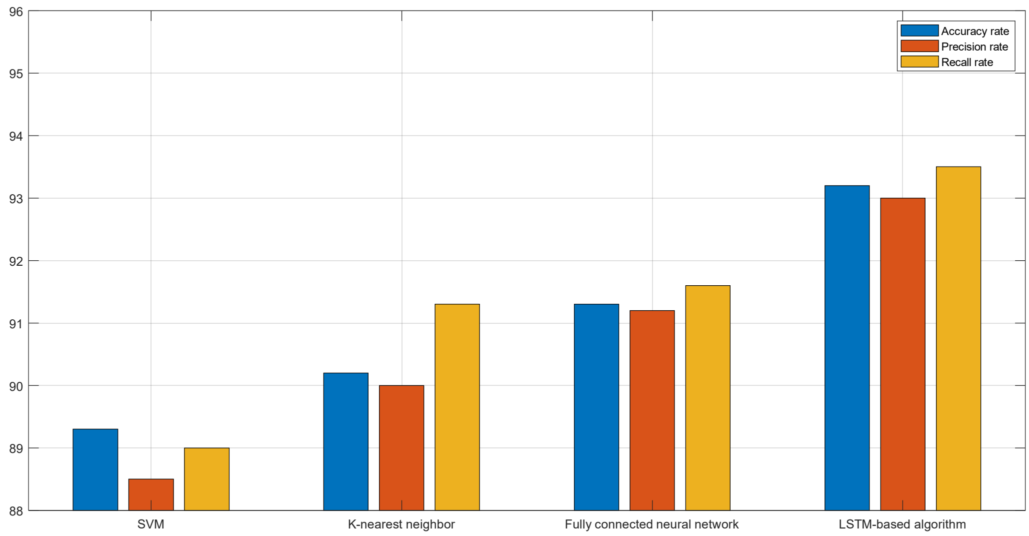

Figure 7: Performance evaluation under user independence

Table 2 below shows the accuracy of each gesture under the condition of user dependence. The accuracy rate is above 96, which means that in most cases, the method proposed in this paper can obtain satisfactory results, especially for simple gestures. But because of gestures' high similarity and the large difference in the angle of the tester's gestures, "complex gesture 5" and "complex gesture 6" are easy to confuse, and the accuracy rate is about 3% lower than that of other gestures.

Table 2: Accuracy of each hand gesture recognition

Gesture	Accuracy rate (%)
Up	96.70
Down	95.03
left	95.73
right	96.04
Draw a circle	96.30
Complex gesture 1	93.30
Complex gesture 2	94.70
Complex gesture 3	94.27
Complex gesture 4	94.87
Complex gesture 5	91.20
Complex gesture 6	91.30
Complex gesture 7	93.27

By recruiting two testers, the trained recognition algorithm is used to recognize gestures under the condition of user independence. Figure 8 below shows the method and directness proposed in this paper under the condition of user independence. The raw sensor data is used for the comparison of gesture recognition. Thanks to the angle feature and displacement feature that are not related to speed, the method proposed in this paper is more stable and robust in the case of user independence.

Real-time evaluation of the method according to the actual use scenario of the handheld motion sensor, the timeliness of the method, is an important factor that affects the user experience.

Table 3 below shows the delay of various methods. The results show that the method proposed in this paper is real-time with satisfactory performance. The delay is even lower than its own fully connected neural network, which is equivalent to the support vector machine. This is because the data redundancy is reduced after feature extraction and the features are obvious, thereby improving the efficiency of the classifier.

Table 3: Delay of each method

Method	Delay(ms)
SVM	122
K-Nearest Neighbor	113
Fully connected neural network	146
LSTM-based algorithm	130

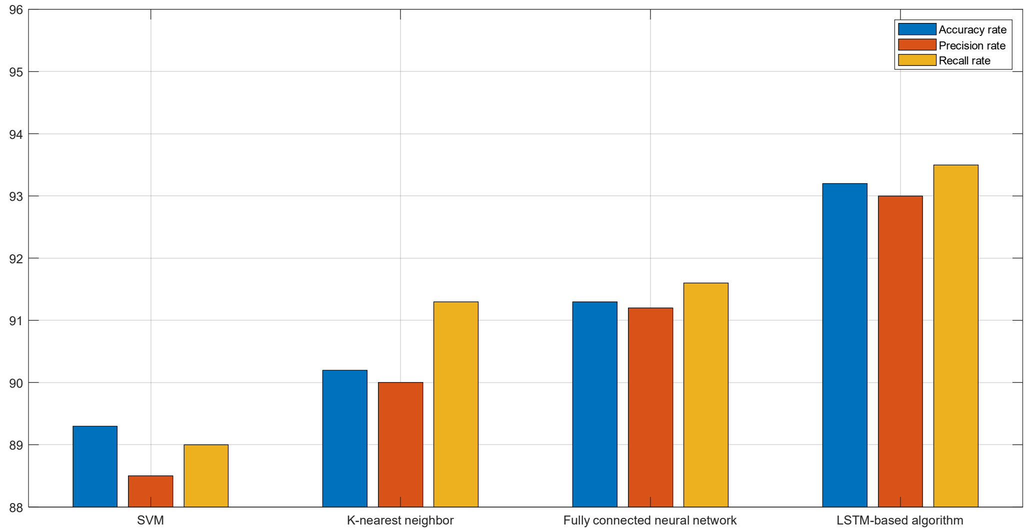

Figure 8: Performance evaluation under user dependence

CONLUSIONS AND RECOMMENDATIONS

This article proposes a real-time recognition algorithm for multi-feature gestures based on LSTM, which extracts motion angle and displacement features according to the characteristics of hand motion, and at the same time extracts frequency domain features through Short-Time Fourier Transform (SFTF) from the sensor data, from the time domain, frequency domain and other multi-dimensional feature extraction to reduce the impact of user differentiated actions. This method is composed of signal acquisition and data preprocessing, feature extraction, data segmentation, and LSTM classifier construction. Context training is used to obtain the final recognition results. Experimental results have shown that for the 12 predefined gestures, the accuracy of the methods proposed in this paper are all above 90%. For independent test data, the accuracy of gesture recognition reaches 93.50%, which is different from the traditional motion sensor-based gesture recognition. Compared to the support vector machine, K-nearest neighbor method and fully connected neural network methods, the accuracy rate has increased by 2%. At the same time, compared to traditional machine learning methods, the time-consumption of this method has decreased significantly. In summary, the proposed method has excellent results when performing gesture recognition.

REFERENCE

Almaslukh B, J. A. (2017). An effective deep autoencoder approach for online smartphone based human activity recognition. *International Joint Conference on Computer Science*, pp. 160-165.

Alsheikh M A, S. A. (2015). Deep Activity Recognition Models with Triaxial Accelerometers. *Computer Science*, 64-78.

Chen L C, W. F. (2013). A Survey on Hand Gesture Recognition. *International Conference on Computer Sciences and Application.* Wuhan, China.

Chen L, L. Y. (2019). Gesture recognition in virtual reality interactive games. *Technological Innovation and Application,* 20, 22-24.

Chu Y, J. Y. (2020). Recognition of Hand Gesture Sequences by Accelerometers and Gyroscopes. *Applied Sciences,* 10(18).

Hammerlar N Y, H. S. (2016). Deep,Convolutional,and Recurrent Models for Human Activity Recognition using Wearables. *Journal of Scientific Computing,* 2(61), 454-476.

Kundu A S, M. O. (2018). Hand Gesture Recognition Based Omnidirectional Wheelchair Control Using IMU and EMG Sensors. *Journal of Intelligent & Robotic Systems* 91, 529–541.

Lane N D, P. G. (2015). DeepEar:robust smartphone audio sensing in unconstrained acoustic environments using deep learning. *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pp. 283-294. New York.

Li Y, S. D. (2014). Unsupervised Feature Learning for Human Activity Recognition Using Smartphone Sensors. *Mining Intelligence and Knowledge Exploration*, 99-107.

Liu Y H, S. L. (2014). 2 D Human Gesture Tracking and Recognition by the Fusion of MEMS Inertial and Vision Sensors. *IEEE Sensors Journal,* 14(4), 1160-1170.

Lu Z, C. X. (2017). A Hand Gesture Recognition Frameworkand Wearable Gesture-Based Interaction Prototype for Mobile Devices. *IEEE Transactionson Human-Machine Systems,* 44(2), 293-299.

Miao Y W, L. J. (2020). Gesture recognition based on joint rotation feature and fingertip distance feature. *Chinese Journal of Computers,* 43(1), 78-92.

Nafea O, A. W. (2021). Sensor-Based Human Activity Recognition with Spatio-Temporal Deep Learning. *Sensors,* 21, 1-20.

Oliver A, M. N. (n.d.). Technology; New Findings from Helwan University Update Understanding of Technology (Smart Healthcare Solutions Using the Internet of Medical Things for Hand Gesture Recognition System). *Journal of Engineering,* 99, 1701-1706.

Oscar D L, M. A. (2013). A Survey on Human Activity Recognition using Wearable Sensors. *IEEE*

Communications Surveys and Tutorial, 15(3), 915-919.

Thomas P, H. N. (2011). Feature Leaning for Activity Recognition in Ubiquitous Computing. *Proceedings of the 22nd International Joint Conference on Artificial Intelligence.* Barcelon.

Vepakomm A P, D. D. (2015). A Wristocracy:Deep Learning on Wrist worn Sensing for Recognition of User Complex Activities. *IEEE Body Sensor Networks Conference.* New York.

Walse K H, D. R. (2016). PCA Based Optimal ANN Classifiers for Human Activity Recognition Using Mobile Sensors Data. *International Conference on ICT for Intelligent Systems*, pp. 429-436. Cham.

Wang A, C. G. (2016). Human Activity Recognition in a Smart Home Environment with Stacked Denoising Autoencoders. *Springer*, 29-40.

Wang J S, C. F. (2012). An accelerometer-based digital pen with a trajectory recognition algorithm for handwritten digit and gesture recognition. *IEEE Transactions on Industrial Electronics,* 59(7), 2998-3007.

Xiong Y, Q. F. (2006). Hand Motion Gesture Frequency Properties and Multimodal Discourse Analysis. *International Journal of Computer Vision,* 69(3), 353-371.

Y, B. (2013). Deep learning of representations: Looking forward. *International Conference on Statistical Language and Speech Processing.* Berlin.

Zhang X, C. X. (2011). A Framework for Hand Gesture Recognition Based on Accelerometer and EMG Sensors. *IEEE Transactionson Systems, Man,and Cybernetics,* 41(6), 1064-1076.

Zhao N, Y. X. (2020). Circulating Nurse Assistant:Non-contact Body Centric Gesture Recognition towards Reducing Iatrogenic Contamination. *IEEE Journal of Bimedical and Health Informatics,* 25, 2305-2316.

Zhou, J. L., H. Z. (2020). Design and implementation of VR piano and gesture recognition based on Leapmotion. *ElectronicTesting,* 21, 5-9.

Zhu T, J. G. (2013). Kinect Application Overview and Development Prospect. *Modern Computer (Professional Edition),* 33(6), 8-11.

Zou, Z., W. Q. (2020). Design of Smart Car Control System for Gesture Recognition Based on Arduino. 2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE): 204-205. Guangzhou, China.