

# Study on gesture recognition based on IIDTW algorithm

**Tian Pei, Chen Guozhen<sup>\*</sup>, Li Nianfeng**

<sup>1</sup>School of information engineering, Communication University of China, Beijing, China  
[e-mail:tianpei@263.net, cheugen@cuc.edu.cn, wawjlnf@163.com]

<sup>\*</sup>Corresponding author: Chen Guozhen

*Received December 28, 2018; revised April 25, 2019; revised June 2, 2019; accepted July 12, 2019;  
published December 31, 2019*

---

## **Abstract**

When the length of sampling data sequence is too large, the method of gesture recognition based on traditional Dynamic Time Warping (DTW) algorithm will lead to too long calculation time, and the accuracy of recognition result is not high. Support vector machine (SVM) has some shortcomings in precision, Edit Distance on Real Sequences (EDR) algorithm does not guarantee that noise suppression will not suppress effective data. A new method based on Improved Interpolation Dynamic Time Warping (IIDTW) algorithm is proposed to improve the efficiency of gesture recognition and the accuracy of gesture recognition. The results show that the computational efficiency of IIDTW algorithm is more than twice that of SVM-DTW algorithm, the error acceptance rate is FAR reduced by 0.01%, and the error rejection rate FRR is reduced by 0.5%. Gesture recognition based on IIDTW algorithm can achieve better recognition status. If it is applied to unlock mobile phone, it is expected to become a new generation of unlock mode.

---

**Keywords:** Gesture password; Android sensor; IIDTW algorithm; Authentication; Mobile phone unlock

## 0. Quote

With the rapid development of information technology in the world, more and more mobile phone users begin to pay more and more attention to the safety and convenience of mobile phone use. This is mainly because there is a large amount of important information stored on the mobile phone, many users bind it with their identity, so the security of the mobile phone is particularly important. In addition, there are many items used by users in today's era, so simplifying the operation steps of a single item has become a strong call (that is, convenience) in today's era. Based on the security and convenience of mobile phone use, this paper studies the topic of mobile phone smart unlock.

As we know, smart phone unlock has become a mainstream trend in today's era, such as TouchID fingerprint recognition technology introduced by iPhone5s [1] and FaceID facial recognition technology introduced by iPhoneX [2]. However, some current smart unlock methods require some expensive chips to be installed on mobile phones, which directly leads to a sharp rise in the cost of mobile phone users in the world. For example, the Chinese market price of mobile phones without fingerprint unlock and facial unlock is about 1,000 yuan, while the market price of mobile phones with any of the above functions is generally higher than 2,000 yuan. In addition, these mainstream smart unlock methods are unreliable because they rely on biometric traits that will eventually be changed[3] or damaged[4]. Therefore, it is of great significance to study a reliable and inexpensive gesture recognition intelligent unlock system.

Based on the above reasons, this paper analyzes and studies the gesture password intelligent unlock system. The main principle of this system is that when a mobile phone user takes his mobile phone to make a specific action, the data collected by the mobile phone sensor is verified by some matching algorithms. If the verification passes, the unlock is completed; if the verification fails, the unlock operation is not performed.

The core of the unlocking system is the matching algorithm, which is mainly the DTW matching method [5]. However, this algorithm cannot meet the requirements for the gesture identification system with high precision. In addition, this algorithm has a low operating efficiency. If it is faced with a large gesture database [6], this disadvantage of this algorithm will be obvious. Last year, we first proposed the concept of gesture password unlocking[7]. At that time, the matching algorithm adopted was a combination of SVM and DTW. However, although this algorithm can perform well in accuracy, it still has great room for improvement in computational efficiency. Considering the processing accuracy and computational efficiency, a gesture recognition system based on IIDTW algorithm is proposed. Let's start with the data objects that the algorithm handles.

## 1. Trajectory data

In this experiment, data of linear acceleration sensor, gyroscope sensor, rotation vector sensor and pressure sensor are collected to describe the running trajectory of user gesture through data of the first three sensors. The algorithm to generate the trajectory data is as follows:

## 1.1 Effective interception of gesture data

Firstly, the source gesture data needs to be intercepted effectively. In this experiment, the whole collection process is to press the button to start the collection and release the button to end the collection. Therefore, at the beginning and the end, the pressure sensor will produce obvious changes. We regard this obvious change as the transition time of the effective data, so we can use the return value of the pressure sensor to intercept the data.

It is advisable to assume that the data set of A pressure sensor is  $D_1, D_2, D_3, \dots, D_n$ . If the KTH value  $D_k$  satisfies condition  $\frac{|D_{k+1} - D_k|}{D_{k-2}} \geq 1\% (k = 1, 2, \dots, \frac{n}{2})$ , data  $D_k$  is considered

invalid; if the KTH value  $D_k$  satisfies condition  $\frac{|D_k - D_{k-1}|}{D_{k-1}} \geq 1\% (k = \frac{n}{2}, \frac{n}{2} + 1, \dots, n)$ ,

data  $D_k$  is considered invalid. This method can effectively eliminate the data that interferes with the data collection this time.

## 1.2 Gesture coordinate transformation

After intercepting gesture data, the vector data of linear acceleration sensor and gyro sensor need to be converted to the earth coordinate system. This paper mainly uses the quaternion method[8] to realize the conversion of gesture data to the earth coordinate system. Now let's give a specific introduction to its implementation.

Quaternion method, its basic form is to use a scalar and three vectors together to represent one data, such as  $q = \lambda + p_1i + p_2j + p_3k$ , if expressed by the quaternion method can be written as  $q = \cos \frac{\theta}{2} + \sin \frac{\theta}{2} n$ , only the following formula is true:

$$\lambda = \cos \frac{\theta}{2}, p_1 = \sin \frac{\theta}{2} \cdot n_x, p_2 = \sin \frac{\theta}{2} \cdot n_y, p_3 = \sin \frac{\theta}{2} \cdot n_z \quad (1)$$

Where  $n = (n_x, n_y, n_z)$ . Now we need to find a transformation matrix  $R(n, \theta)$  that allows  $v$  to rotate  $\theta$  around axis  $n$  and get a new vector  $v'$ , which is  $R(n, \theta)v = v'$ . Now, let's break down vector  $v$  into two components that are parallel and perpendicular to axis  $n$ , which we'll call  $v_{\parallel}$  and  $v_{\perp}$ . It's easy to see that  $v_{\parallel}$  doesn't change as it rotates around axis  $n$ , and  $v_{\perp}$  changes as it rotates around axis  $n$ , so  $v' = v_{\parallel} + v_{\perp}'$ . Now construct the intermediate variable  $w$  to satisfy  $|w| = |v_{\perp}|$ , and  $w = v_{\parallel} \times v_{\perp}$ , then formula (2) is true:

$$\begin{cases} v' = v_{\parallel} + v_{\perp}' \\ v_{\perp}' = v_{\perp} \cos \theta + w \sin \theta \\ w = n \times v_{\perp} \\ v_{\parallel} = (v \cdot n)n \end{cases} \quad \text{where } n \text{ is a vector of unit length} \quad (2)$$

By derivation, we can get:

$$v' = (v - (v \cdot n)n) \cos \theta + (v \times n) \sin \theta + (v \cdot n)n \quad (3)$$

Now, take the three basis vectors  $p_1 = (1, 0, 0)^T, p_2 = (0, 1, 0)^T, p_3 = (0, 0, 1)^T$  of the geodetic coordinate system and rotate around the coordinate axis  $n = (n_x, n_y, n_z)$  of the mobile phone coordinate system respectively, that is, let the vector  $v$  be equal to

$p_1, p_2, p_3$  respectively, and substitute into formula (3), the following results can be obtained:

$$R(n, \theta) = \begin{bmatrix} n_x^2(1 - \cos \theta) + \cos \theta & n_x n_y(1 - \cos \theta) + n_z \sin \theta & n_x n_z(1 - \cos \theta) - n_y \sin \theta \\ n_x n_y(1 - \cos \theta) - n_z \sin \theta & n_y^2(1 - \cos \theta) + \cos \theta & n_y n_z(1 - \cos \theta) + n_x \sin \theta \\ n_x n_z(1 - \cos \theta) + n_y \sin \theta & n_y n_z(1 - \cos \theta) - n_x \sin \theta & n_z^2(1 - \cos \theta) + \cos \theta \end{bmatrix} \quad (4)$$

Then, according to formula (1), the relation between  $R(n, \theta)$  and  $\lambda, p_1, p_2, p_3$  can be obtained by using semi-angular formula  $1 - \cos \theta = 2 \sin^2 \frac{\theta}{2}, \cos \theta = 1 - 2 \cos^2 \frac{\theta}{2}, \sin \theta = 2 \sin \frac{\theta}{2} \cos \frac{\theta}{2}$  and the property values[0]<sup>2</sup> + values[1]<sup>2</sup> + values[2]<sup>2</sup> + values[3]<sup>2</sup> = 1 of the rotating vector sensor, as shown in formula (5).

$$R(n, \theta) = \begin{bmatrix} \lambda^2 + p_1^2 - p_2^2 - p_3^2 & 2(p_1 p_2 + \lambda p_3) & 2(p_1 p_3 - \lambda p_2) \\ 2(p_1 p_2 - \lambda p_3) & \lambda^2 - p_1^2 + p_2^2 - p_3^2 & 2(p_2 p_3 + \lambda p_1) \\ 2(p_1 p_3 + \lambda p_2) & 2(p_2 p_3 - \lambda p_1) & \lambda^2 - p_1^2 - p_2^2 + p_3^2 \end{bmatrix} \quad (5)$$

Of which,  $\lambda, p_1, p_2, p_3$  is from values[3], values[0], values[1], values[2] of the rotating vector sensor. Considering the differences of mobile phone manufacturers,  $\lambda$  here is set as  $\sqrt{1 - \text{values}[0]^2 - \text{values}[1]^2 - \text{values}[2]^2}$ . In this way, formula (5) can be used to convert the initial data of the linear acceleration sensor and the gyroscope sensor into the data in the inertial coordinate system.

### 1.3 Trajectory data generation

Now we introduce the algorithm of gesture trajectory data generation. Now we have linear acceleration data in the earth coordinate system ( $a_{xi}, a_{yi}, a_{zi}, i = 1, 2, \dots, n$  represents the  $i$ TH data), gyro angular velocity data ( $v_{xi}, v_{yi}, v_{zi}, i = 1, 2, \dots, n$  represents the  $i$ TH data), and the working frequency of the sensor is known to be  $f$ . If  $t_{n-1}$  represents the previous time of the NTH data at time and  $t_n$  represents the time of the NTH data at time, then  $t_{n-1} = (n-1)/f$  and  $t_n = n/f$ .

Now, if we project this data onto three axes, and we analyze it, we find that there are four known variables on each axis, for example, on the X-axis, there are known variables  $v_{x(n-1)}, a_{x(n-1)}, v_{xn}, a_{xn}$ . Now, assuming that the x coordinate value of the (n-1) data is  $x_{n-1}$ , if we want to find the coordinate of the next data,  $x_n$ , we need to find the displacement distance  $s$  on the X-axis.

Now modeling with the above conditions, for a variable speed curve motion, it has 4 known variables. Considering that the modeling method of cubic polynomial can guarantee certain accuracy and not be too complicated, this experiment adopts the mathematical model based on the speed and acceleration of cubic polynomial, and the specific implementation is as follows.

Assuming that the velocity  $v_{xn}(t)$  in the NTH time interval satisfies the expression:  $v_{xn}(t) = px_{1n}t^3 + px_{2n}t^2 + px_{3n}t + px_{4n}$ , then the acceleration  $a_{xn}(t)$  in this time satisfies the

expression:  $a_{xn}(t)=3px_{1n}t^2+2px_{2n}t+px_{3n}$ , substitute the above four known variables into these two expressions, and get:

$$\begin{cases} 3px_{1n}t_{n-1}^2 + 2px_{2n}t_{n-1} + px_{3n} = a_{x(n-1)} \\ 3px_{1n}t_n^2 + 2px_{2n}t_n + px_{3n} = a_{xn} \\ px_{1n}t_{n-1}^3 + px_{2n}t_{n-1}^2 + px_{3n}t_{n-1} + px_{4n} = v_{x(n-1)} \\ px_{1n}t_n^3 + px_{2n}t_n^2 + px_{3n}t_n + px_{4n} = v_{xn} \end{cases} \quad (6)$$

In order to solve the coefficient  $px_{1n}$ ,  $px_{2n}$ ,  $px_{3n}$ ,  $px_{4n}$ , it needs to be converted into matrix form first, as shown in formula (7),

$$\left[ \begin{array}{cccc|c} 3t_{n-1}^2 & 2t_{n-1} & 1 & 0 & a_{x(n-1)} \\ 3t_n^2 & 2t_n & 1 & 0 & a_{xn} \\ t_{n-1}^3 & t_{n-1}^2 & t_{n-1} & 1 & v_{x(n-1)} \\ t_n^3 & t_n^2 & t_n & 1 & v_{xn} \end{array} \right] \quad (7)$$

A solution of four unknown numbers can be obtained by solving the formula (7),

$$\begin{cases} px_{1n} = f^2(a_{x(n-1)} + a_{xn}) + 2f^3(v_{x(n-1)} - v_{xn}) \\ px_{2n} = f(2-3n)(a_{x(n-1)} + a_{xn}) + 3f^2(2n-1)(v_{x(n-1)} + v_{xn}) \\ px_{3n} = na_{x(n-1)}(3n-2) + a_{xn}(3n-1)(n-1) + 6fn(n-1)(v_{x(n-1)} - v_{xn}) \\ px_{4n} = -\frac{n(n-1)}{f}[na_{x(n-1)} + a_{xn}(n-1)] - n^2v_{x(n-1)}(2n-3) + v_{xn}(2n^3 - 3n^2 + 1) \end{cases} \quad (8)$$

Then the displacement expression for the interval is:

$$s_{xn}(t) = \frac{1}{4} px_{1n}t^4 + \frac{1}{3} px_{2n}t^3 + \frac{1}{2} px_{3n}t^2 + px_{4n}t + C_n \quad (9)$$

Where, the value interval of t is  $[t_{n-1}, t_n]$ , and the time t is substituted into the formula (9) to get

$$C_{xn} = s_{x(n-1)}(t-1) - \left[ \frac{(n-1)^4}{4f^4} px_{1(n-1)} + \frac{(n-1)^3}{3f^3} px_{2(n-1)} + \frac{(n-1)^2}{2f^2} px_{3(n-1)} + \frac{n-1}{f} px_{4(n-1)} \right] \quad (10)$$

By using formula (10), the x-direction coordinate value  $s_{xn}(n)$  corresponding to each movement of gesture data can be calculated step by step by iterative method.

Similarly, we can calculate the coordinates  $s_{yn}(n)$  and  $s_{zn}(n)$  corresponding to the other two axes. Therefore, the inertial trajectory coordinate corresponding to each movement of gesture data can be represented by  $(s_{xn}(n), s_{yn}(n), s_{zn}(n))$ , which is the object solved by matching algorithm.

### 1.4 Gesture data completion and fitting

The converted gesture source data set may be denoted as  $AX'i(n)$ ,  $AY'i(n)$ ,  $AZ'i(n)$ , where  $i=1, 2, \dots, 10$  represents the  $i$ th data collection, and  $n=1, 2, 3, \dots$  represents the length of data obtained by each data collection. In addition, due to the different speed of data collection, the length of these 10 data sets will be different, which will affect data fitting. Therefore, the above multiple sets of data (and the data to be tested) should be uniformly supplemented to a certain length L to make it meet  $L = \max_{0 < i < 11} \{ \text{length}(AX'i(n)), \text{length}(AY'i(n)), \text{length}(AZ'i(n)) \}$ , where  $\text{length}(AX'i(n))$ ,  $\text{length}(AY'i(n))$  and  $\text{length}(AZ'i(n))$  represent

the length of the data cleaned for the  $i$ th time. The method of completion is cubic spline interpolation[9], which is specifically defined and implemented as follows:

**Definition:** suppose there are interpolation nodes on  $[a, b]$ , where  $a=x_1 < x_2 < \dots < x_n = b$  corresponds to  $y_1, y_2, \dots, y_n$ . If the function  $S(x)$  satisfies  $S(x_j)=y_j(j=1, 2, \dots, n)$ ,  $S(x)$  is a polynomial no higher than the third degree in each interval  $[x_j, x_{j+1}]$ . When  $S(x)$  has the second continuous derivative on  $[a, b]$ , then  $S(x)$  is called the cubic spline interpolation function. We need to determine the unique cubic polynomial on each subinterval  $[x_j, x_{j+1}]$  on  $[a, b]$ , which is now denoted as

$$S_j(x)=a_jx^3+b_jx^2+c_jx+d_j, (j=1, 2, \dots, n-1) \quad (11)$$

Where  $a_j, b_j, c_j, d_j$  are the coefficient to be solved and should satisfy the following expression:

$$S(x_j)=y_j, S(x_{j-0})=S(x_{j+0}), S'(x_{j-0})=S'(x_{j+0}), S''(x_{j-0})=S''(x_{j+0}), (j=2, \dots, n-1) \quad (12)$$

Since the number of conditions in formula (12) is  $n+3(n-2)=4n-6$ , it can be seen from formula (11) that there are  $4n-4$  coefficients to be solved. Therefore, it is necessary to find 2 known conditions. Here, the result expression of the coefficient to be solved is given by taking the first derivative value at the known endpoint as an example:

Assuming  $S''_j(x)=M_j, S''_{j+1}(x)=M_{j+1}$ , and denoting  $h_j=x_{j+1}-x_j$ , the expression of  $S(x)$  can be deduced as follows:

$$S(x)=\frac{(x_{j+1}-x)^3}{6h_j}M_j+\frac{(x-x_j)^3}{6h_j}M_{j+1}+(y_j-\frac{M_j}{6}h_j^2)\frac{x_{j+1}-x}{h_j}+(y_{j+1}-\frac{M_{j+1}}{6}h_j^2)\frac{x-x_j}{h_j} \quad (13)$$

In formula (13), the solution of coefficient  $M_j, M_{j+1}$  to be solved satisfies formula (14):

$$\begin{bmatrix} 2 & 1 & & & & & \\ \mu_2 & 2 & \gamma_2 & & & & \\ & \ddots & \ddots & \ddots & & & \\ & & \mu_{n-1} & 2 & \gamma_{n-1} & & \\ & & & 1 & 2 & & \end{bmatrix} \begin{bmatrix} M_1 \\ M_2 \\ \vdots \\ M_{n-1} \\ M_n \end{bmatrix} = \begin{bmatrix} \beta_1 \\ d_2 \\ \vdots \\ d_{n-1} \\ d_n \end{bmatrix} \quad (14)$$

$$\text{Where } \mu_j = \frac{h_{j-1}}{h_{j-1}+h_j}, \gamma_j = 1-\mu_j, d_j = 6\left(\frac{y_{j+1}-y_j}{h_j} - \frac{y_j-y_{j-1}}{h_{j-1}}\right)\frac{1}{h_{j-1}+h_j}, (j=2,3,\dots,n-1) \quad (15)$$

$$\beta_1 = \frac{6}{h_1}\left(\frac{y_2-y_1}{h_1} - y_1\right), \beta_n = \frac{6}{h_{n-1}}\left(y_n - \frac{y_n-y_{n-1}}{h_{n-1}}\right)$$

$M_j, M_{j+1}$  in formula (13) can be obtained by solving formula (14), and  $h_j=x_{j+1}-x_j$  in formula (13), so each subinterval  $[x_j, x_{j+1}](j=1, 2, \dots, n-1)$  can uniquely determine a cubic polynomial.

After three spline interpolation, these ten sets of data may be called as  $AX''_i(n), AY''_i(n), AZ''_i(n)$ , where  $i=1, 2, \dots, 10$  represents the  $i$ th time of data collection, and  $n=1, 2, 3$  represents the length of each piece of data after spline interpolation (all equal to length  $L$ ).

So far, the processing of trajectory data has been completed.

## 2. Traditional DTW algorithm

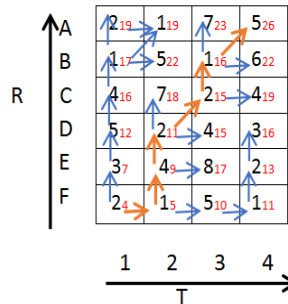
Assuming that there is a standard reference template  $R$  ( $M$ -dimensional vector), let's call it  $R = \{R(1), R(2), \dots, R(M)\}$ , where each component can be a number or a smaller component. There is a test template  $T$  ( $N$ -dimensional vector) to be verified, which can be written as  $T = \{T(1), T(2), \dots, T(N)\}$ .  $M$  is not necessarily equal to  $N$ , but the dimensions of each component are the same. The optimal path of  $R$  and  $T$  can be obtained by using the DTW algorithm.

Suppose the distance from one square  $(i-1, j-1)$  or  $(i-1, j)$  or  $(i, j-1)$  to the next square  $(i, j)$ , if adjacent is denoted as  $d(i, j)$ , if diagonal is denoted as  $2d(i, j)$ , then the minimum distance from the first square to the current square is expressed as  $g(i, j)$ , as shown in formula (16) :

$$g(i, j) = \min\{g(i-1, j) + d(i, j), g(i-1, j-1) + 2d(i, j), g(i, j-1) + d(i, j)\} \tag{16}$$

For example,  $R$  is ABCDEF (6-dimensional vector),  $T$  is 1234 (4-dimensional vector), and its distances are shown in figure 1:

The black number in figure 1 represents the distance between  $R$  and  $T$  vectors. According to formula (1), the distance required for each route can be calculated step by step and the shortest path can be found. Such as this example assumes that the  $g(0, 0) = 0$ , then  $g(1, 1) = g(0, 0) + 2d(1, 1) = 0 + 2 * 2 = 4$ ,  $g(2, 1) = g(1, 1) + d(2, 1) = 4 + 3 = 7$ ,  $g(1, 2) = g(1, 1) + d(1, 2) = 4 + 1 = 5$ , extrapolate out every other distance  $g(i, j)$  value, as shown in figure 1 in red Numbers. The blue arrow is then used to represent the direction of the minimum value source, and finally the shortest path is found through the backtracking method [10], as shown in the path connected by the brown arrow in the Fig. 1.



**Fig. 1.** Getting the best path for  $R$  and  $T$

However, this traditional DTW method still needs to be improved, and it needs to perform traversal calculation [11] on the whole rectangular region. In fact, what we need to study is whether  $R$  and  $T$  can achieve a relatively high degree of confidence [12], so as to judge whether the data to be tested matches the original data. In this way, we will not use those regions with lower confidence. Therefore, you can consider transforming traversal regions.

### 3. IIDTW algorithm

For the  $M$  - dimensional reference template  $R$  and  $N$  - dimensional test template  $T$ , this experiment requires a similarity of more than 95% before a successful match is considered. According to the DTW algorithm, if  $R$  and  $T$  are completely similar, the optimal path should be the line segment passing through the origin  $(0, 0)$  and the end point  $(N, M)$ . Therefore, we only need to study the data points on both sides of this line segment [13]. The arc region is adopted here, and the specific implementation method is shown in Fig. 2:

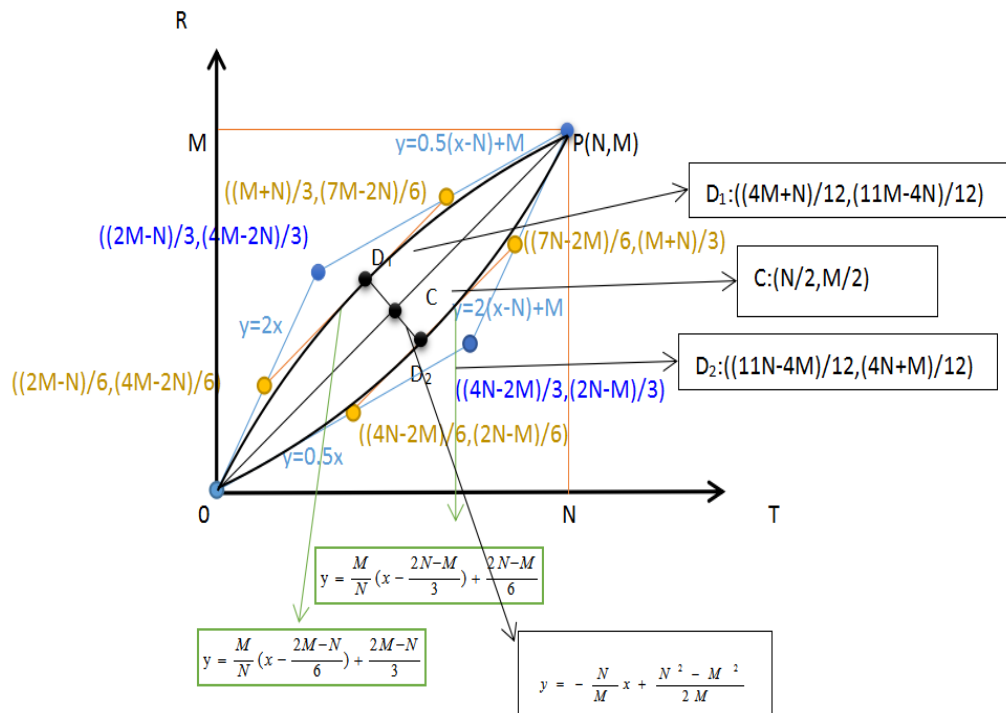


Fig. 2. IIDTW algorithm with arc region

First, draw a diamond with the above line segment as the diagonal edge, as shown in Fig. 2 in blue line for area (in this case, the slopes of the adjacent diamond sides are 2 and 1/2, respectively), and then calculate the equation for each side, as shown in figure 2 in blue expression and then calculate the coordinates of the other two endpoints of diamond, as shown in figure 2 in blue coordinates.

Next, go through the middle points of each diamond and draw line segments  $y = \frac{M}{N} * (x - \frac{2M-N}{6}) + \frac{2M-N}{3}$  and  $y = \frac{M}{N} * (x - \frac{N-2M}{3}) + \frac{2N-M}{6}$  parallel to line segment  $y = \frac{M}{N} * x$ , as shown in the yellow line segment in figure 2. In this way, through the point  $(0, 0)$ , point  $(\frac{2M-N}{6}, \frac{4M-2N}{6})$ , point  $(\frac{2M-N}{3}, \frac{4M-2N}{3})$ , point  $(\frac{M+N}{3}, \frac{7M-2N}{6})$ , point  $(N, M)$ , point  $(\frac{7M-N}{6}, \frac{M+N}{3})$ , point  $(\frac{4N-2M}{3}, \frac{2N-M}{3})$ , point  $(\frac{4N-2M}{6}, \frac{2N-M}{6})$ , point  $(0, 0)$  of hexagon area was built.



Next, take line segment OP as the string, and make the center of the circle go through the straight line CD<sub>1</sub> of the middle point D<sub>1</sub> of arc OP and the middle point C of OP, and draw arc OD<sub>1</sub>P. In this way, the radius R<sub>1</sub> corresponding to arc OD<sub>1</sub>P can be deduced to

satisfy expression  $(R - \frac{1}{12}\sqrt{41(M^2 + N^2) - 80MN})^2 = R^2 - \frac{1}{4}(M^2 + N^2)$ , so

$$R_1 = \frac{\sqrt{41(M^2 + N^2) - 80MN}}{24} + \frac{3(M^2 + N^2)}{2\sqrt{41(M^2 + N^2) - 80MN}} \quad (17)$$

Similarly, draw arc OD<sub>2</sub>P, the radius R<sub>2</sub> corresponding to arc OD<sub>1</sub>P can be deduced to

satisfy expression  $(R - \frac{1}{12}\sqrt{41(M^2 + N^2) - 80MN})^2 = R^2 - \frac{1}{4}(M^2 + N^2)$ , so

$$R_2 = \frac{\sqrt{41(M^2 + N^2) - 80MN}}{24} + \frac{3(M^2 + N^2)}{2\sqrt{41(M^2 + N^2) - 80MN}} \quad (18)$$

In this way, the values of the transverse coordinates of the center O<sub>1</sub> and O<sub>2</sub> corresponding to our two arcs satisfy the expression :

$$x^2 + N \cdot \frac{M^2 - N^2}{M^2 + N^2} \cdot x + \frac{N^4 + 4M^4 - 3M^2N^2 - 4M^2R^2}{4(M^2 + N^2)} = 0$$

. After the final solution, the results of the circular center coordinate x<sub>1,2</sub> are as follows:

$$x_{1,2} = \frac{576\sqrt{41(M^2 + N^2) - 80MN} \times N(N^2 - M^2) \pm 48\sqrt{F}}{2 \times 576\sqrt{41(M^2 + N^2) - 80MN} \times (M^2 + N^2)}$$

$$F = 11443M^4N^4 + 1681N^6M^2 - 6560N^5M^3 - 12173M^6N^2 - \dots$$

$$13120M^5N^3 + 39520M^7N - 21935M^8 + \dots$$

$$3888\sqrt{41(M^2 + N^2) - 80MN} \times (N^2M^6 + N^4M^4) + \dots$$

$$1296\sqrt{41(M^2 + N^2) - 80MN} \times (N^6M^2 + M^8) \quad (19)$$

After solving the x<sub>1,2</sub>-coordinate of the center of the circle x, substitute the linear equation:

$$y = -\frac{N}{M}x + \frac{N^2 - M^2}{2M} \quad (20)$$

You get y<sub>1,2</sub> at the center. In this way, the expression of the curved OD<sub>1</sub>PD<sub>2</sub> region enclosed by two arcs can be expressed by the following formula:

$$(x - x_1)^2 + (y - y_1)^2 \leq \frac{41(M^2 + N^2) - 80MN}{576} + \frac{9(M^2 + N^2)^2}{4 \times [41(M^2 + N^2)^2 - 80MN]}$$

$$(x - x_2)^2 + (y - y_2)^2 \leq \frac{41(M^2 + N^2) - 80MN}{576} + \frac{9(M^2 + N^2)^2}{4 \times [41(M^2 + N^2)^2 - 80MN]} \quad (21)$$

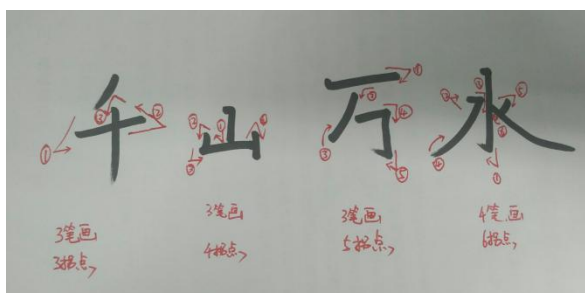
Then the arc region OD<sub>1</sub>PD<sub>2</sub> is the region that we will eventually traverse. Finally, in this region, the running efficiency of DTW algorithm can be greatly improved by using the DTW algorithm for traversal calculation.

## 4. Experimental Result

### 4.1 The Data Collection

The gesture data used in this paper is collected by lenovo zuk pro2 mobile phone. The gesture data collected is converted into quaternion in the form of database, data fitting and IIDTW authentication. Finally, the Euclidean distance obtained by IIDTW is analyzed and the feedback results are displayed on the mobile phone interface.

According to the general password security rule [14], the more complex the password, the higher the password security, that is, the higher the password accuracy. According to this theory, we studied four Chinese gestures: "千", "山", "万" and "水". The strokes of these four characters are relatively simple (3, 4) and the number of inflection points is less (For an incremental increase of 3, 4, 5, 6). As shown in the Fig. 3:



**Fig. 3.** Schematic diagram of the number of inflection points of three strokes Chinese characters

The data collected in this experiment mainly include 20 times of data collected by experimenter A and 5 times of data collected by experimenter B and C. Among the 20 data collected by experimenter A, there were 10 times of data as training data, and another 10 times of data as the data to be verified for lawful matching. The data collected 10 times by experimenters B and C were used as the unmatched data to be verified. Now a total of 30 times the total data, of which 20 times for validating data, each data contains "千", "山", "万", "水" four kinds of situations, and each situation including 10 data (three acceleration data, three angular velocity data, three rotation vector data, a pressure data), only 6 to validate data (three the quaternion method of transformation of acceleration data, three of the quaternion method converting angular velocity data), as a result, a total of  $20 \times 4 \times 6 = 480$  pending data were collected.

## 4.2 The Data Analysis

The total time required for data acquisition by DTW algorithm and IIDTW algorithm is now measured by using the time function, as shown in Fig. 4. According to Fig. 4, it can be seen that the operation efficiency of IIDTW algorithm is more than doubled than that of SVM-DTW algorithm or DTW algorithm.

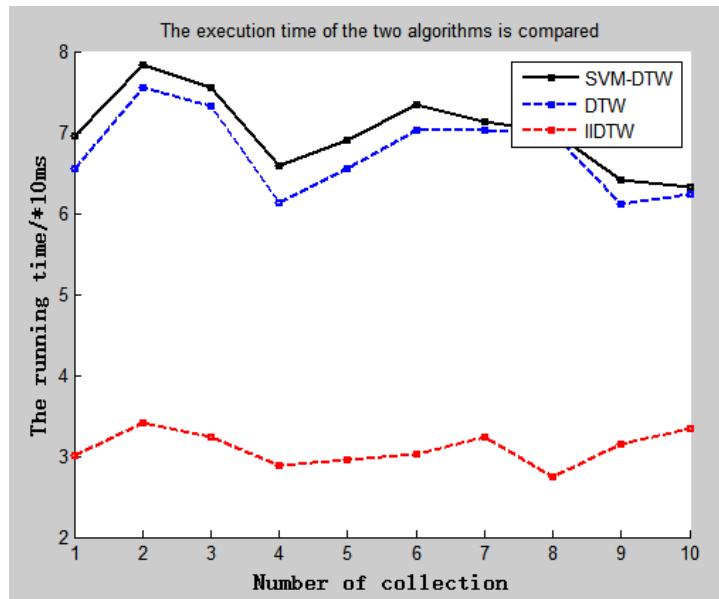


Fig. 4. Comparison of data acquisition time among the three algorithms

By using the training data of experimenter A to generate A matching template, another 10 times of the data to be verified by experimenter A were conducted for legitimate authentication experiments, and 10 times of the data to be verified by experimenter B and C were conducted for illegal authentication experiments. The experimental results of Euclidean distance are obtained by IIDTW authentication algorithm, as shown in Table 1.

Table 1. Experimental results of authentication for different types of gestures

Similarity Distance	Times	Legal certification experiment (Euclidean distance)									
		1	2	3	4	5	6	7	8	9	10
	"千" gesture data	24.98	24.13	20.10	18.22	22.01	17.36	16.42	19.87	20.32	22.67
	"山" gesture data	17.59	22.59	25.72	24.93	23.11	19.87	25.71	20.01	22.11	24.99
	"万" gesture data	18.23	19.34	18.65	25.44	19.35	25.36	26.99	24.55	24.97	19.65
	"水" gesture data	25.31	19.87	19.00	24.65	26.41	16.59	25.47	23.14	22.22	26.01
Similarity Distance	Times	Illegal authentication experiment (Euclidean distance)									
		1	2	3	4	5	6	7	8	9	10
	"千" gesture data	23.15	36.57	29.63	26.84	21.79	38.96	36.77	25.03	37.10	30.09
	"山" gesture data	36.77	25.78	39.99	31.12	34.83	25.03	29.71	43.23	35.99	25.72
	"万" gesture data	36.45	29.00	41.23	44.06	50.36	41.00	33.23	36.45	49.66	35.98
	"水" gesture data	61.33	54.86	39.88	65.21	46.81	35.74	36.44	38.97	46.93	58.34

Now compare the two authentication experiments of unified gestures in the above data and use the MATLAB drawing tool to get the following results:

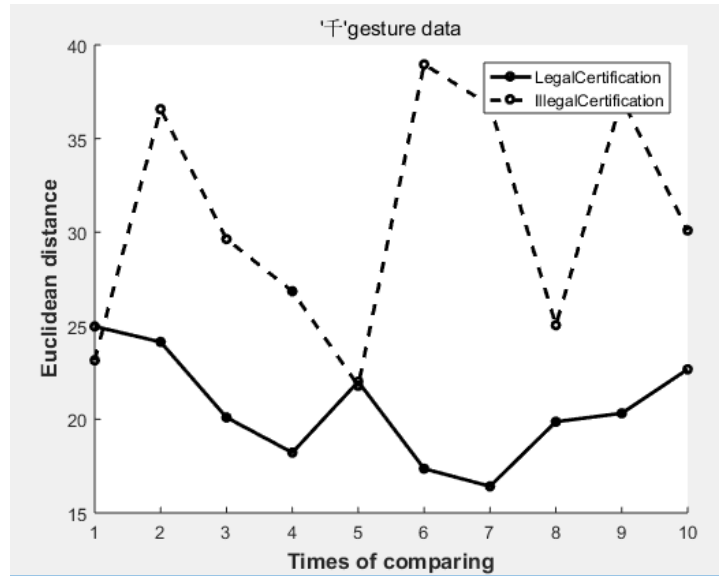


Fig. 5. Comparison of "千" gesture data

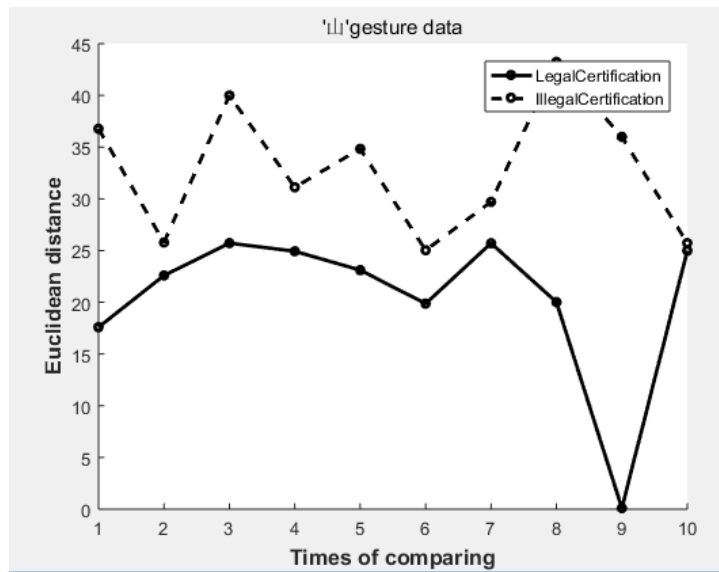


Fig. 6. Comparison of "山" gesture data

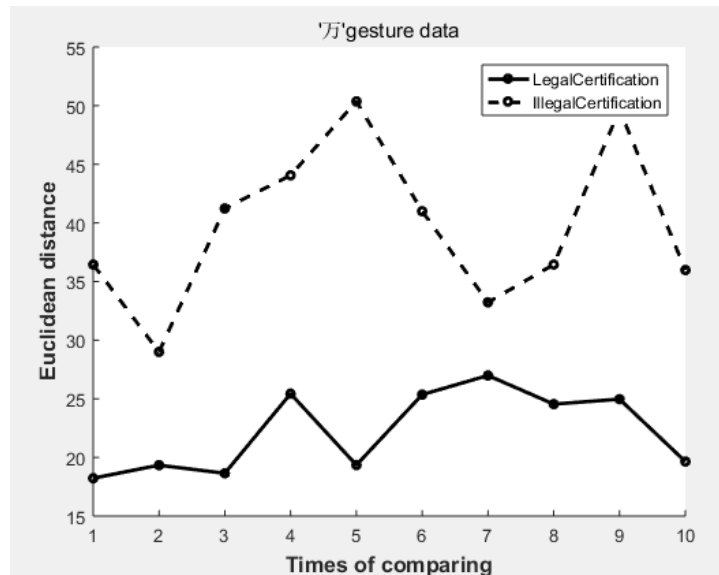


Fig. 7. Comparison of "万" gesture data

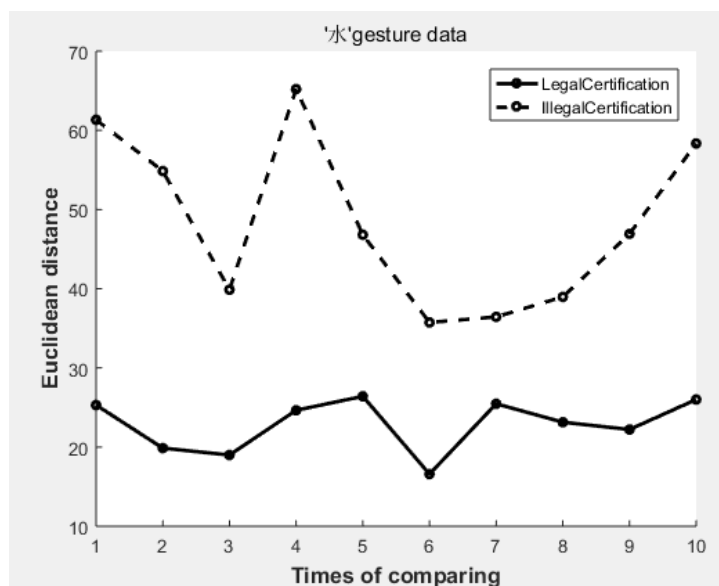


Fig. 8. Comparison of "水" gesture data

According to the comparison results of the four groups above, there is no clear dividing line between legal data and illegal data of "thousand" and "mountain", while there is a clear dividing line between legal data and illegal data of "wan" and "water". Therefore, for these gestures with distinct boundaries, a specific similar distance value can effectively distinguish legitimate data from illegal data.

And there is a rule for these different types of gesture data: the more inflection points, the more complex the gesture, the greater the gap between legal authentication and illegal authentication, which means that the more complex the gesture, the higher the feasibility of identity authentication. In terms of behavior habits[15], certain users make similar movements with their phones, and the more complex a gesture is, the more important it is.

### 4.3 Experimental Analysis

In the previous section, we know that the more complicated the gesture is, the higher the feasibility of authentication is. We know that among the four gestures of "千", "山", "万" and "水", the simplest is "千" data, while the most complicated is "水". As can be seen from the comparison chart of "万" gesture data, there is a clear dividing line between legal data and illegal data of "万" gesture at five inflection points. Given the practicality, the gesture data should be simpler if we use the gesture password to unlock it. As we know, the eight principles of Yong [16] is the pen rule of Chinese calligraphy. The essence of the eight strokes of the word "yong" is actually eight strokes. Wang xizhi, a great calligrapher in the eastern jin dynasty, believed that all Chinese characters could be written as long as they could write "yong". So in this experiment, additional 20 volunteers were collected to verify the experimental results of a relatively simple "永" gesture data. These 20 volunteers were firstly input the gesture data and generated the training template, and then each of them performed 10 matching authentication operations on the 20 training data. Thus, the total number of legal certifications is  $20 \times 10 = 200$ , and the total number of illegal certifications is  $10 \times 190 = 1,900$ . The experimental results are shown in **Table 2**:

**Table 2.** Results of 20 volunteers

	User 1	User 2	User 3	User 4	User 5	User 6	User 7	User 8	User 9	User 10
legal certification failure times	0	0	1	0	0	0	0	0	0	0
illegal certification success times	0	0	0	0	0	1	0	0	1	0
	User 11	User 12	User 13	User 14	User 15	User 16	User 17	User 18	User 19	User 20
legal certification failure times	1	0	0	0	0	0	0	0	0	1
illegal certification success times	0	0	1	0	0	0	0	0	0	1

It can be seen from table 2 that this scheme can achieve good results for the certification of most users, and thus the error acceptance rate [17] of the experiment is  $FAR = \text{illegal certification success times} / \text{illegal certification total times} = 4/1900 = 0.21\%$ , and the error rejection rate of the experiment is  $FRR = \text{legal certification failure times} / \text{legal certification total times} = 3/200 = 1.5\%$ . The similar effect was compared to the 0.22% FAR and 2% FRR when SVM-DTW algorithm was used. However, it can be seen from section 4.2 that its operation efficiency is much higher than that of SVM-DTW algorithm.

## Conclusion

In this paper, we mainly studied some functional sensors of Android phones[18], which can be used for identity authentication through some processing operations of the sensors embedded in these phones. In this paper, data collected by various sensors embedded in smart phones are analyzed to extract and analyze gesture features more comprehensively. In this paper, the linear acceleration sensor is used to represent the linear motion characteristics of the hidden password of a gesture, which is collected by a gesture operation. The angular velocity characteristics of the hidden password of gestures collected by a user performing a gesture operation are represented by the gyroscope sensor to indicate the set of phone inclination of a particular gesture. The finger force characteristics of the hidden password of gestures collected by a user performing a gesture operation with a pressure sensor are used to represent the strength set of mobile phone usage of a specific user due to his/her habit. The rotation vector sensor is used to transform the mobile phone sensor into geodetic coordinate system to eliminate the influence of the mobile phone orientation.

In this paper, an improved interpolation dynamic time warping, IIDTW, is proposed, which can efficiently and accurately calculate the Euclidean distance between the reference template and the test template. Through the analysis of gestures with different degrees of complexity, this paper finds that the higher the degree of complexity of gestures, the stronger the feasibility of identity authentication. And the experiment verifies that the "万" data of five inflection points can satisfy the identification and authentication of identity. So we can tag each of us with a specific tag for identification by making simple gestures with our phones.

## References

- [1] Yang Zhenghui, Li Duo, "The Design and Implementation of a Fingerprint Lock Based on AS602," *Chinese Journal of Electron Devices*, 39(02), 495-500, 2016.  
[Article \(CrossRef Link\)](#)
- [2] Deng, L.Y., Dong Liang Lee, Yi-Jen Liu, "Face Recognition Lock," in *Proc. of 2013 International Conference on IT Convergence and Security (ICITCS)*, 2013.  
[Article \(CrossRef Link\)](#)
- [3] Rodolf Haraksim, Javier Gallbally, Laurent Beslay, "Fingerprint Growth Model for Mitigating the Ageing Effect on Children's Fingerprints Matching," *Pattern Recognition*, 88, 614-628, 2019.  
[Article \(CrossRef Link\)](#)
- [4] Javad Khodadoust, Ali Mohammad Khodadoust, Xiong Li, Saru Kumari, "Design and Implementation of a Multibiometric System Based on Hand's Traits," *Expert System with Applications*, 97, 303-314, 2018.  
[Article \(CrossRef Link\)](#)
- [5] Li Zhengxin, Zhang Fengming, Li Kewu, Zhang Xiaofeng, "Index Structure for Multivariate Time Series under DTW Distance Metric," *Journal of Software*, 25(03), 560-575, 2014.  
[Article \(CrossRef Link\)](#)
- [6] Cheng Xueqi, Jin Xiaolong, Wang Yuanzhuo, Guo Jiafeng, Zhang Tieying, Li Guojie, "Survey on Big Data System and Analytic Technology," *Journal of Software*, 25(09), 1889-1908, 2014.  
[Article \(CrossRef Link\)](#)
- [7] Li Nianfeng, "Research of The Gesture Authentication Method Based on Android and Its Realization," *Beijing: School of Electronic Information Engineering, Communication University of China*, 37-38, 2018.

- [8] Zhang Jinxue, Zhang Ming, Li Yuanyuan, “Quaternion - Based Attitude Control of Fixed-wing Micro Air Vehicle,” *Computer Measurement & Control*, 20(07), 1851-1854, 2012. [Article \(CrossRef Link\)](#)
- [9] Luo Tianhong, Ma Li, “Path Planning for Spider Biomimetic Robot Leg based on Cubic Polynomial Interpolation Function with Midpoints,” *Mechanical Drive*, 41(07), 1-6, 2017. [Article \(CrossRef Link\)](#)
- [10] Chen Yue, Ding Lingwei, Wu Hongxuan, Liu Guohai, Liu Hui, “Power Quality Signal Reconstruction of Backtracking-Based Self-Adaptive Matching Pursuit Method,” *Journal of Electronic Measurement and Instrumentation*, 31(05), 731-738, 2017. [Article \(CrossRef Link\)](#)
- [11] Li Jing, Song Jing, Long Pengcheng, Liu Hongfei, Jiang Ping, “Parallel Computing Method of Monte Carlo Criticality Calculation Based on Bi-directional Traversal,” *Nuclear Techniques*, 38(06), 95-98, 2015. [Article \(CrossRef Link\)](#)
- [12] Liu Shushu, Liu An, Liu Guanfeng, Li Zhixu, Zhao Lei, Zheng Kai, “A Privacy Preserving Framework for Efficient Computation of Trajectory Similarity,” *Journal of East China Normal University(Natural Science)*, 2015(05), 154-161+171, 2015. [Article \(CrossRef Link\)](#)
- [13] Zhao Lei, Zhou Yimin, “A Gesture Recognition Based on Improved DTW-IMP Algorithm,” *Software Guide*, 16(11), 12-15+21, 2017. [Article \(CrossRef Link\)](#)
- [14] Lu Gang, Nie Xuyun, Qin Zhiguang, Hou Chuanyong, “Cryptanalysis of a Multivariate Public Key Cryptosystem,” *Journal of University of Electronic Science and Technology of China*, 47(02), 242-246, 2018. [Article \(CrossRef Link\)](#)
- [15] Li Qingwu, Ma Yunpeng, Zhou Yan, Zhou Liangji, “Method of Writer Identification Based on Curvature of Strokes,” *Journal of Chinese Information Processing*, 30(05), 209-214, 2016. [Article \(CrossRef Link\)](#)
- [16] Yan Lindong, “Combined with the “Eight Principles of Yong” Analysis the Rationality of the Regular Script Writing Action,” *China Academy of Art*, 2017. [Article \(CrossRef Link\)](#)
- [17] Xianjing MENG, Xiaoming XI, Gongping YANG, Yilong YIN, “Finger vein recognition based on deformation information,” *Science China(Information Sciences)*, 61(05), 156-170, 2018. [Article \(CrossRef Link\)](#)
- [18] Huang Yiming, Lei Hang, Zhou Rui, Sang Nan, “Activity and Holding Mode Recognition Using Multiple Sensors,” *Journal of University of Electronic Science and Technology of China*, 46(02), 399-406, 2017. [Article \(CrossRef Link\)](#)





**Pei Tian:**He received the B.S. degree and M.S. degree from Northwestern Polytechnical University, Shanxi, China.He received his PhD degree in Beihang University, Beijing, China.He is now a professor in the Communication University of China.His research interests include artificial intelligence, software development and testing.



**Guozhen Chen:**He is currently pursuing a master degree at the School of Information Engineering in Communication University of China, Beijing, China.His current research interests include gesture recognition and matching algorithm.



**Nianfeng Li:**He received his M.S. degree at the School of Information Engineering in Communication University of China, Beijing, China, in 2018.His current research interests include gesture recognition and pattern recognition.