Dynamic Gesture Recognition based on Kernel Density Estimation of Fingertip Angle Set

Haibo Lin [a], Shengbin Wang [b] and Yi Zhang [c]

Engineering Research and Development Center for Information Accessibility, Chongqing University of Posts and Telecommunication, Chongqing 400065, China

[a]844038217@qq.com, [b]1357503397@qq.com, [c]1905613719@qq.com

Abstract

A fast and table recognition algorithm combining gesture trajectory and gesture types was proposed for the lack of robustness and real-time ability of dynamic gesture recognition in the complex environment. Gesture region was obtained using Kinect sensor in experiment space and accurate gesture region was obtained by a method of variable parameter quickly. Gesture 3D trajectory extraction and recognition was based on the depth data stream. Real-time gesture types recognition based on kernel density estimation sequence of fingertip angle set template matching algorithm was implemented. Experiments demonstrate that the method of gesture segmentation is fast and accurate. Robustness and real-time ability of gesture trajectory recognition and type recognition is better.

Keywords

Dynamic Gesture Recognition, 3D Trajectory Recognition, FAC-KDES, Template Matching.

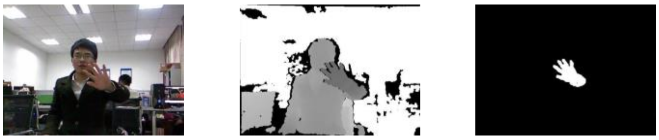
1. Introduction

Gesture interaction based on machine vision is one of the key technology of intelligent human-computer interaction[1-4]. With the development of computer technology, the application of real-time dynamic gesture recognition in the complex environment is wide[2,5-7]. In the dynamic gesture recognition, three problems need to be solved[2,6,8,9]: gesture segmentation, gesture feature extraction and gesture recognition. In the thesis, the variable parameter algorithm can be used to split the accurate gesture, which lays the foundation for the gesture feature extraction and recognition. The real-time template matching algorithm in gesture recognition has good robustness.

Different gesture types are represented by the combination of different fingers and the finger is the key to gesture[2,6,10]. In reference [1], the concept of FEMD is introduced by extracting the relative distance curve feature of gesture edge and palm and gesture type is effectively judged by template matching. In reference [5], gesture type is analyzed by Kalman filter tracking gesture trajectory and "U" curve. But not each gesture has smooth "U" shape, which limites the number of gesture types, the division requirement is too high. In reference [9], gesture edge gradient decomposition feature is calculated and the recognition effect is well. Reference [1,9] require precise gesture segmentation and the marker-less segmentation method requires too much. Reference [10,11] use the radius-based convex hull decomposition[12] algorithm to judge gesture types well with the template matching after much calculation of the shape concave degree[12] and complex gesture skeleton extraction algorithm. In addition, reference [1,9,10,11] has high computational complexity and real-time recognition is poor. Therefore, dynamic gesture recognition method based on kernel density estimation of fingertip angle set is proposed and the validity of the method is proved.

2. Accurate gesture segmentation

Reference [1,2,6,9,10,11] use Kinect to obtain RGB image, depth shaded image and gesture binary image, as shown in Fig. 1.



(a) RGB image (b) Depth shaded image (c) Gesture binary image

Fig. 1 Three gesture images

As shown in Fig. 1 (c), the gesture segmentation based on the depth information is not accurate with wrist and the variable parameter algorithm can be used to split the accurate gesture. As shown in Fig. 2.

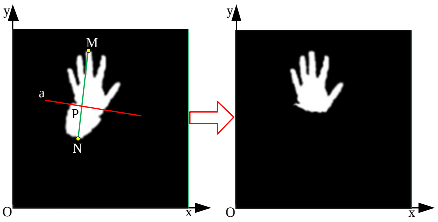


Fig. 2 Accurate gesture segmentation

3. Gesture feature extraction

3.1 Gesture 3D trajectory stream extraction

Dynamic gesture 3D trajectory stream extraction is the real-time response of a fixed point coordinates[2], the target gesture center of gravity coordinates and the average gray value flow are selected as the 3D trajectory coordinate stream. Experimental space coordinate system is shown in Fig. 3, gesture trajectory point coordinates is shown in formula (1).

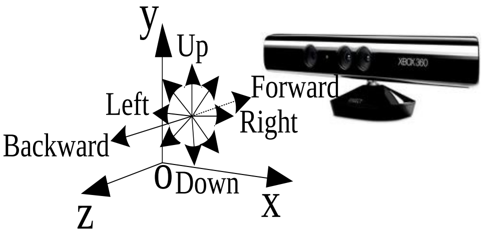
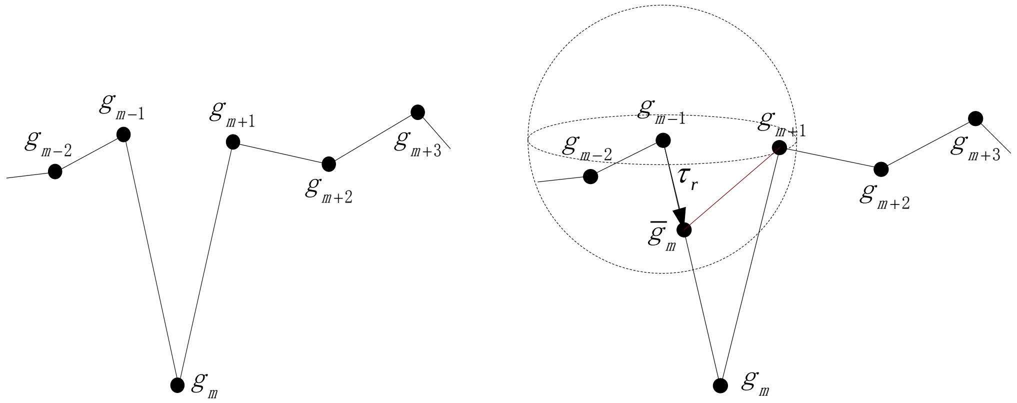


Fig. 3 Experimental space coordinate system

$$(\bar{x}_t,\bar{y}_t,\bar{z}_t)=(\frac{\sum_{i=1}^{m}\sum_{j=1}^{n} jH_3(i,j)}{S_{H_3}},m-1-\frac{\sum_{i=1}^{m}\sum_{j=1}^{n} iH_3(i,j)}{S_{H_3}},\frac{\sum_{i=1}^{m}\sum_{j=1}^{n} G(i,j)H_3(i,j)}{S_{H_3}}),S_{H_3}=\sum_{i=1}^{m}\sum_{j=1}^{n} H_3(i,j). \quad (1)$$

In the case of sudden fluctuation in gesture trajectory in Fig. 4 (a), the point g_m will cause the inaccuration of trajectory. Therefore, the ball radius critical limit method for processing is proposed. As shown in Fig. 4 (b), the distance of point g_m and point g_{m-1} exceeds the critical radius τ_r of the ball. The intersection $\bar{g}_m$ of the space line $|g_{m-1}g_m|$ and the current spherical surface (the point g_{m-1} is the ball center and τ_r is the ball radius) is the output for the next gesture trajectory point. Point $\bar{g}_m$ space coordinates calculation is formula (2), $|g_{m-1}g_m|$ is the distance between the two points.

$$\bar{g}_m=g_{m-1}+\frac{\tau_r}{|g_{m-1}g_m|}(g_m-g_{m-1}). \quad (2)$$



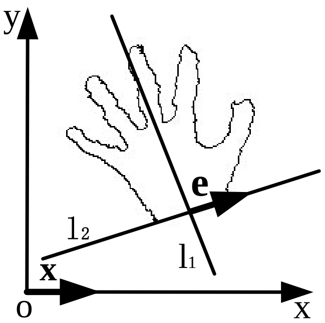
(a) Abnormal gesture trajectory (b) Processing of gesture trajectory ball radius critical limit method

Fig. 4 Abnormal processing of gesture 3D trajectory

3.2 Sequence extraction of kernel density estimation of fingertip angle set

3.2.1 Fingertip angle set

According to the usual gesture type, as shown in Fig. 5, l_1 represents the line of arm, l_2 represents the dividing line of wrist and palm, $l_1 \perp l_2$, a unit vector ***e*** is existing always, at the time, the vector ***e*** is defined as the cutting vector of the wrist.



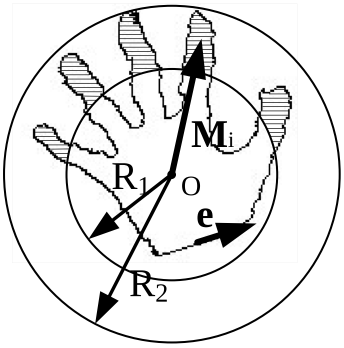


Fig. 5 Definition of wrist cutting vector Fig. 6 Definition of fingertip angle set

Fig. 6 is the gesture model without wrist, ***e*** is the wrist cutting vector, point O is palm center point when distance shaded image maximum shaded value is got in the gesture image distance transformation, R_2 is the radius of gesture circumcircle with circle center O.

$$R_1 = \alpha R_2, \alpha_1 < \alpha < \alpha_2. \tag{3}$$

In the formula (3), Choice of α_1 and α_2 needs to meet that all finger parts of any gesture are not connected. Vector M_i represents vector of each pixel where point O to the fingertip. M_i is the fingertip vector.

Define the fingertip angle set S as:

$$S = \{\theta_i \mid \theta_i = \arccos(e \bullet \frac{M_i}{|M_i|}), \quad 0 < \theta_i < \pi, \quad 0 < i \le N_1, \quad i \in N^*\}. \tag{4}$$

N_1 is the number of pixels in the fingertip area, S is called as the fingertips angle set.

3.2.2 Kernel density estimation

Kernel density estimation[13,14] is used to estimate the unknown density function and it is one of the nonparametric test methods. It is proposed by Rosenblatt and Emanuel Parzen and it is called as the Parzen window.

For data $x_1, x_2, ..., x_N$, the kernel density estimate is in the following form.

$$\hat{f}(x)=\frac{1}{N}\sum_{i=1}^{N}K_h(x-x_i)=\frac{1}{Nh}\sum_{i=1}^{N}K(\frac{x-x_i}{h}),h>0. \quad (5)$$

In the formula (5), $K(\bullet)$ is the kernel function and it satisfies (1)$\int_{-\infty}^{+\infty}|K(u)|du<+\infty$, (2)$\lim_{|u|\to\infty}uK(u)=0$, (3)$K(-u)=K(u)$, (4)$\int_{-\infty}^{+\infty}K(u)d(u)=1$. $K(u)\geq 0$ is existing for arbitrary u, h is bandwidth. The choice of kernel function and the bandwidth determines the degree of the effect of the estimation. The choice of kernel function depends on the difference in the contribution of each sample point being assigned to the density according to the distance. Common kernel function is uniform kernel, Gaussian kernel, cosine kernel, exponential kernel and so on. In general, h has a great influence on the smoothness of the density estimation distribution. It is important to choose the suitable h.

3.2.3 Sequence of kernel density estimation of fingertip angle set

Accurate wrist cutting vector ***e*** in gesture segmentation is not easy to be obtained, so the unit vector ***e*** is got as in Fig. 5, which may result in θ_i horizontal drift on the angle axis, making it difficult to the subsequent correlation. Therefore, S in the angle $[\theta_{\min},\theta_{\max}]$ is interval normalized to $[0,1]$. As shown in formula (6).

$$\theta_i'=\frac{\theta_i-\theta_{\min}}{\theta_{\max}-\theta_{\min}},i=1,2,3,...,N_{angle}. \quad (6)$$

In the formula (6), N_{angle} represents the number of element of a fingertip angle set sample. It is equal to the N_1 in formula (4). In view of the nature of fingertip angle set, the Gaussian kernel is chosen as the kernel function.

$$K(u)=\frac{1}{\sqrt{2\pi}}e^{-\frac{u^2}{2}}. \quad (7)$$

The kernel density estimation is deduced as the following formula (8).

$$\hat{f}(\theta')=\frac{1}{\sqrt{2\pi}N_{angle}h}\sum_{i=1}^{N_{angle}}e^{-\frac{(\theta'-\theta_i')^2}{2h^2}},i=1,2,3,...,N_{angle},h>0. \quad (8)$$

Then, the kernel density estimation curve $\hat{f}(\theta')$ of the fingertip angle set of the sample gesture is obtained, which is similar to the relative distance curve between gesture edge and palm center in reference [1]. The characteristic curve in reference [1] is not smooth enough and the correlation matching method is not easy to be used. The minimal FEMD (H, Tc) algorithm with high complexity is used for template matching.

To further reduce the matching complexity, sequence $\hat{f}(\theta')$ of kernel density estimation of fingertip angle set is obtained by uniform sampling.

$$\hat{u}(k)=\hat{f}(\theta')\varepsilon(k),\varepsilon(k)=\begin{cases}1,k=\dfrac{i}{N_{sam}}, & i\in N^*,k\in[0,1].\\ 0,\text{其他}\end{cases} \quad (9)$$

In formula (9), N_{sam} is the number of uniform sampling.

For the different gesture types of the same topological structure[10,11], the FAC-KDES feature is significantly different due to the difference of the proportion between the fingertip distance and each fingertip. The sampling of the number of fixed points on the corresponding normalized interval $[0,1]$ of interval $[\theta_{\min},\theta_{\max}]$ also enlarges the detail differences of the FAC-KDES feature. Moreover, the

FAC-KDES feature fully satisfies gesture scaling and rotation invariance and it is robust to similar skeletal gesture types[1].

4. Dynamic gesture recognition

The dynamic gesture recognition method framework is shown in Fig. 7.

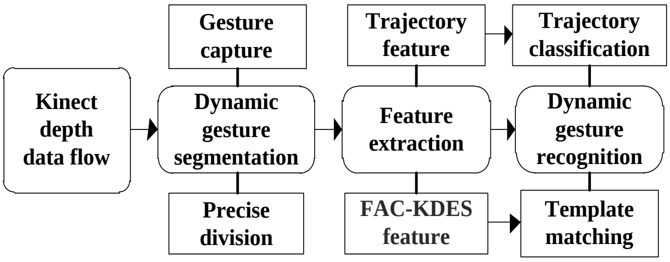


Fig. 7 Dynamic gesture recognition method framework

4.1 Dynamic target area extraction

In gesture type recognition, gesture shape feature is the key. In order to reduce the computation, the square-shaped gesture sub-block with center $(\bar{x}_t,\bar{y}_t)$ and side length $2r_t+\varepsilon$ is intercepted in the binary image corresponding to the matrix $\boldsymbol{D}$. As shown in Fig. 8, r_t represents the radius of gesture circumscribed circle and ε is a small positive odd number. Define $\mathbf{E}_{(2r_t+\varepsilon)\times(2r_t+\varepsilon)}$ as the binary matrix corresponding to the sub-block region. The conversion method of $\mathbf{D}$ to $\mathbf{E}$ is shown in formula (10), R represents the subscripts set of gesture sub-block region in $\mathbf{D}$.

$$\mathbf{E}(p,q)=\mathbf{D}(i,j),(i,j)\in R,\begin{cases} p=i-(m-1-\bar{y}_t-\dfrac{2r_t+\varepsilon-1}{2}) \\ q=j-(\bar{x}_t-\dfrac{2r_t+\varepsilon-1}{2}) \end{cases}. \qquad (10)$$

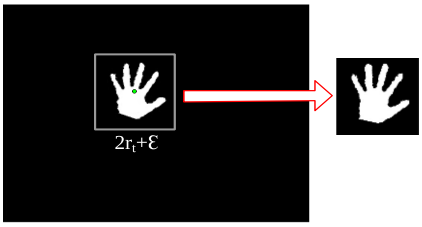


Fig. 8 Gesture target interception

4.2 Dynamic gesture 3D trajectory recognition

The trajectory recognition includes coordinate stream mode and direction mode. The coordinate flow mode directly reads 3D trajectory coordinate flow and direction mode converts the coordinate stream into 10 direction information in Fig. 4.

In direction mode, T is the time period of direction information output and the $T/\Delta t$ secondary direction information is determined. The displacement vector is shown below.

$$\Delta\mathbf{p}_{NT+n\Delta t}=\mathbf{p}_{NT+(n+1)\Delta t}-\mathbf{p}_{NT+n\Delta t}=(\Delta\bar{x}_{NT+n\Delta t},\Delta\bar{y}_{NT+n\Delta t},\Delta\bar{z}_{NT+n\Delta t}),$$
$$N=0,1,2,3,...;n=0,1,2,...,T/\Delta t-1. \qquad (11)$$

Define $\mathbf{m}_{NT+n\Delta t}=(\Delta\bar{x}_{NT+n\Delta t},\Delta\bar{y}_{NT+n\Delta t})$, $\varphi_{NT+n\Delta t}$ is the angle of the same direction to $\mathbf{m}_{NT+n\Delta t}$ after $\mathbf{x}$ counterclockwise rotation, $\varphi_{NT+n\Delta t}\in[0,2\pi)$.

φ is divided into eight intervals $[\frac{15}{8}\pi,2\pi)\cup[0,\frac{\pi}{8}),[\frac{\pi}{8},\frac{3}{8}\pi),...,[\frac{13}{8}\pi,\frac{15}{8}\pi)$ $(U_1,U_2,...,U_8)$ and the direction is judged as "right", "top right", ..., "lower right". In the time $NT+n\Delta t$ to $NT+(n+1)\Delta t$, the direction decision process is shown in Fig. 9, Γ_{static} is the static threshold,

$N_1, N_2..., N_8, N_{static}, N_{backw}, N_{forw}$ is the direction counting in T . At the end of the cycle T , define the direction of $\max\{N_1, N_2..., N_8, N_{static}, N_{backw}, N_{forw}\}$ as the output direction.

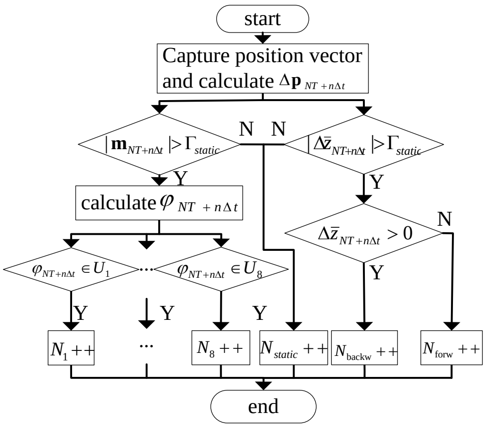

start
Capture position vector and calculate $\Delta\mathbf{p}_{NT+n\Delta t}$
$|\mathbf{m}_{NT+n\Delta t}|>\Gamma_{static}$
N
N
$|\Delta\bar{z}_{NT+n\Delta t}|>\Gamma_{static}$
Y
Y
calculate $\varphi_{NT+n\Delta t}$
$\Delta\bar{z}_{NT+n\Delta t}>0$
N
$\varphi_{NT+n\Delta t}\in U_1$
...
$\varphi_{NT+n\Delta t}\in U_8$
Y
Y
Y
Y
N_1++
...
N_8++
$N_{static}++$
$N_{backw}++$
$N_{forw}++$
end

Fig. 9 Direction determination process in Δt

4.3 4.3 Gesture types template matching

The template matching method is used to identify gesture types by comparing the FAC-KDES average cross-correlation coefficient between the input gesture and the template library sample[15]. Average correlation coefficient is shown below.

$$\bar{\rho}(H_{input}, T_c) = \frac{1}{n}\sum_{i=1}^{n}\rho(\hat{u}_{H_{input}}(k), \hat{u}_{T_{ci}}(k)), c = 1,2,..., N\,. \tag{12}$$

In the formula (12), N is the number of gesture types, n is the number of each gesture type template, T is the template gesture.

$$\rho(\hat{u}_{H_{input}}(k), \hat{u}_{T_{ci}}(k)) = \frac{Cov(\hat{u}_{H_{input}}(k), \hat{u}_{T_{ci}}(k))}{\sqrt{D(\hat{u}_{H_{input}}(k))}\sqrt{D(\hat{u}_{T_{ci}}(k))}}\,. \tag{13}$$

The gesture types discrimination pseudo code is as follows:

```
$if\,(\max(\bar{\rho}(H_{input},T_1),\bar{\rho}(H_{input},T_2),...,\bar{\rho}(H_{input},T_N))\ge\Gamma_{mat})$
{
  $c=\max_c(\bar{\rho}(H_{input},T_1),\bar{\rho}(H_{input},T_2),...,\bar{\rho}(H_{input},T_N));$
  $H_{output}=$"Gesture $c$";
}
$else$
  $H_{output}=$"Unrecognition";
```

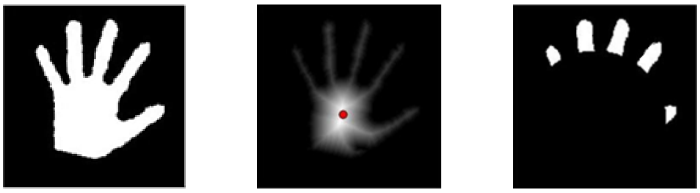
Γ_{mat} is the matching threshold, $\Gamma_{mat}\in(-1,1)$, H_{output} represents the output gesture types.

The method can distinguish $C_5^1+C_5^2+C_5^3+C_5^4+C_5^5=31$ gesture types. Moreover, $31\times10=310$ dynamic gesture types are distinguished in the combination of the 10 kinds of direction information in direction mode.

5. Comparison and analysis of dynamic gesture recognition experiment

5.1 Gesture type recognition analysis

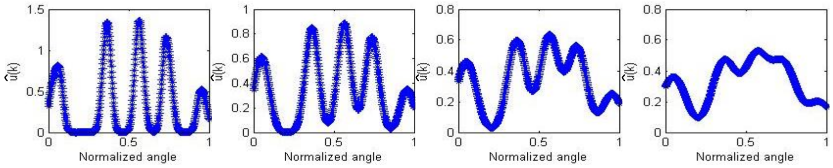
For the target gesture in Fig. 8, the gesture sub-block is obtained from formula (10) with $2r_t + \varepsilon = 161$. Find the palm point by the obtained distance shaded image from the distance transformation. The fingertip part is obtained with $\alpha = \frac{2}{3}$ in formula (3), as shown in Fig. 10.



(a) Gesture sub-block (b) Distance shaded image (c) Fingertip part

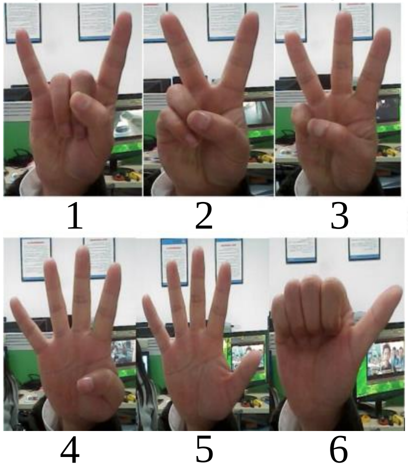
Fig. 10 Fingertip area extraction processing

Fingertip samples in Fig. 10 (c) take different bandwidth h and the number of sampling points $N_{sam} = 500$. The obtained FAC-KDES is in Fig. 11. With the increase of bandwidth h, $\hat{u}(k)$ is more smooth, the difference of different gesture types feature is smaller. It losts the meaning of gesture feature when $h > 0.2$. In addition, each five sample $H_{c1}, H_{c2}, ..., H_{c5}$ in the six general gesture types in Fig. 12 are selected. The $\bar{\rho}(\hat{u}_{H_{c1}}(k), \hat{u}_{H_{c2}}(k), ..., \hat{u}_{H_{c5}}(k)) - h$ image is shown in Fig. 13. h being selected too large or too small is not conducive to the matching of internal gesture. In general, in order to minimize the difference of the internal gesture and maximize the difference of external gesture, $h = 0.1$ is the best value for $\hat{u}(k)$.



h=0.05 h=0.1 h=0.15 h=0.2

Fig. 11 FAC-KDES change with the change of h in the same sample



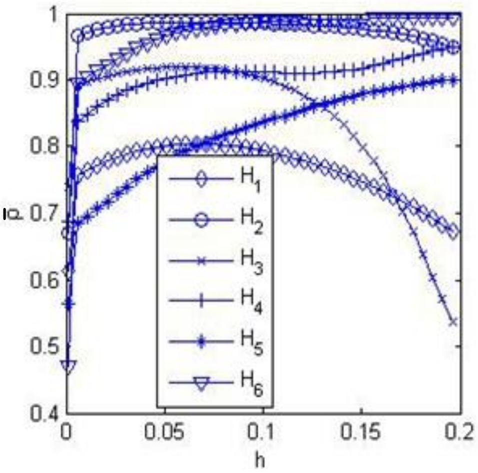


Fig. 12 Six general gesture types Fig. 13 Six general gesture sample $\bar{\rho}$ change with the change of h

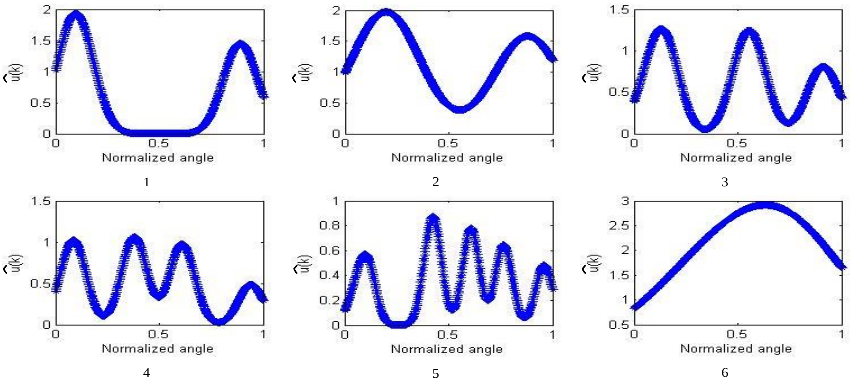


Fig. 14 FAC-KDES of six general gesture

Table 1.Gesture template matching result

H_{input} $\bar{\rho}$ T_c	Library 1	Library 2	Library 3	Library 4	Library 5	Library 6
1	0.8703	0.7752	0.2797	-0.0294	-0.2489	-0.7972
2	0.6808	0.9601	-0.1015	-0.3711	-0.5308	-0.6458
3	0.4532	0.0665	0.9043	0.0351	-0.0462	-0.2136
4	0.0817	-0.3321	0.2522	0.8886	0.5643	-0.0558
5	-0.1888	-0.4214	0.1379	0.4303	0.8219	0.2386
6	-0.8825	-0.6918	-0.1071	-0.1863	0.2592	0.9397
Fist (Undefined)	-0.1849	-0.1023	0.1588	0.0997	0.0156	0.2058

Table 2.Comparison of dynamic gesture direction recognition rate

Method	Right	R-top	Top	L-top	Left	L-down	Down	R-down	Forward	Backward	Average
Method [6]	0.938	0.865	0.914	0.884	0.944	0.937	0.924	0.905	no	no	0.912
Our method	0.950	0.890	0.960	0.870	0.960	0.900	0.970	0.910	0.970	0.980	0.9360

Table 3.Comparison of gesture types recognition rate in different methods

Method	1	2	3	4	5	6	Average
Thresholding Decomposition+FEMD in[3]	0.87	0.85	0.90	0.91	0.90	0.92	0.8917
Near-convex Decomposition+FEMD in[3]	0.95	0.96	0.93	0.92	0.95	0.93	0.9400
Method [6]	0.92	0.90	0.90	0.91	0.93	0.90	0.9100
Our method	0.89	0.90	0.95	0.95	0.94	0.98	0.9350

For the six general gesture in Fig. 12, define $n=5$ in the formula (12) and the corresponding FAC-KDES is shown in Fig. 14. Identified gesture samples and sampling fist gesture are input system and the correlation test data is shown in Table 1. According to the method, H_{input} is positively correlated with the corresponding template. The closer the value of $\bar{\rho}(H_{input},T_c)$ is to 1, the more likely that these two samples are the same gesture type and the closer the value is to -1, the less likely that these two samples are the same gesture type. Combing with the data in Table 1, generally the matching threshold $\Gamma_{mat} = 0.75$, which can be a good way to identify gesture type and distinguish unsampling gesture type.

5.2 Dynamic gesture recognition and comparison

In gesture trajectory recognition direction mode, the determination period $T = 1s$, the determination times $T / \Delta t = 5$ in the period and the static threshold $\Gamma_{static} = 10$. Repeat the test in 10 directions for 100 times. The results comparison between the experiment results and direction recognition results in reference [10] is shown in Table 2. In reference [10], eight static gesture recognition results are used to represent eight directions information in the plane. In the thesis, dynamic gesture direction is used to describe the 10 directions in the space and the interaction is more convenient and intuitive without complex static gesture recognition calculation. The method in "up", "down", "left" and "right" directions is obviously superior to reference [10] and the method has also "forward" and "backward" directions with high recognition rate， which increases the average recognition rate.

In the same conditions, 8-bit binary gesture image samples with 600 161 * 161 pixel (100 gesture from each types in Fig. 12) are counted with the recognition rate and time in reference [1], reference [10] and the method. The results are shown in Table 3 and Table 4. It can be seen from Table 3 that the method has no obvious advantages to similar gesture (such as gesture 1 and gesture 2) and the recognition rate of other gesture is higher than reference [10]. The average recognition rate is higher than the two methods except for the decomposition of the convex hull in reference [1] and the robustness is better. In addition, it can be seen from Table 4 that the gesture type recognition algorithm takes 9.6%, 2.0% and 52.0% of the other three methods.

Table 4.Time-consuming comparison of gesture type recognition in different methods

Method	Total time	Average time
Thresholding Decomposition+FEMD in[3]	542.04s	0.9034s
Near-convex Decomposition+FEMD in[3]	2611.70s	4.3529s
Method [6]	100.14s	0.1669s
Our method	52.06s	0.0867s

Take the six gesture types in Fig. 12 and combine the six directions in the direction mode to identify dynamic gesture for 100 times. The recognition accuracy is shown in Table 5. It can be seen from the table that the average recognition rate of dynamic gesture recognition is 91.67%, which satisfies the robustness of dynamic gesture recognition.

Table 5. Accuracy of six dynamic gesture recognition

	Right-1	Top-2	Left-3	Down-4	Forward-5	Backward-6	Average
Accuracy	0.88	0.89	0.92	0.94	0.93	0.94	0.9167

6. Conclusion

In the thesis, the robustness of dynamic gesture recognition is improved by combining gesture 3D trajectory and gesture type. Accurate gesture segmentation method is based on the variable parameter. In Kinect scene, the 3D trajectory is extracted and recognized based on depth data and any gesture shapes can be recognized based on FAC-KDES feature extraction. The experiment results show that the calculation is reduced obviously for dynamic gesture recognition. Dynamic gesture recognition in the thesis has theoretical and practical significance for real-time human-computer interaction.

Acknowledgements

Chongqing Science and Technology Commission Foundation under Grant (CSTC2015jcyjBX0066)

References

[1] Z. Ren, J. Yuan, J. Meng, et al: Robust Part-based Hand Gesture Recognition Using Kinect

Sensor J]. Multimedia, IEEE Transactions on, Vol. 15 (2013) No.5, p.1110-1120.

[2] S S. Rautaray, A. Agrawal: Vision Based Hand Gesture Recognition for Human Computer Interaction: a Survey J]. Artificial Intelligence Review, Vol. 43 (2015) No.1, p.1-54.

[3] E. Keogh, L. Wei, X. Xi, et al: Supporting Exact Indexing of Arbitrarily Rotated Shapes and Periodic Time Series under Euclidean and Warping Distance Measures J]. The VLDB Journal—The International Journal on Very Large Data Bases, Vol. 18 (2009) No.3, p.611-630.

[4] Y. Chen, Y. Zhang: Research on Human-Robot Interaction Technique Based on Hand Gesture Recognition J]. Robot, Vol. 31 (2009) No.4, p.351-356.

[5] G F. He, S K. Kang, W C. Song, et al: Real-time Gesture Recognition Using 3D Depth Camera[C]// IEEE, International Conference on Software Engineering and Service Science. IEEE (July 15-17, 2011), p.187-190.

[6] J. Han, L. Shao, D. Xu, et al: Enhanced Computer Vision with Microsoft Kinect Sensor: A review J]. IEEE transactions on cybernetics, Vol. 43 (2013) No.5, p.1318-1334.

[7] S B. Xiang, G D. Su, X L. Ren, et al: Embedded Implementation of Real-time Finger Interaction System J]. Guangxue Jingmi Gongcheng(Optics and Precision Engineering), Vol. 19 (2011) No.8, p.1911-1920.

[8] F. Liu, G. Ding, S. Li, et al: ICONDENSATION-Based Hand Tracking and Gesture Recognition with Color and Depth Cues J]. Journal of Beijing Institute of Technology, Vol. 27 (2007) No.12, p.1069-1072.

[9] S S. Rautaray, A. Agrawal: Real Time Hand Gesture Recognition System for Dynamic Applications J]. International Journal of Ubicomp, Vol. 3 (2012) No.1, p.21.

[10] S. Qin, X. Zhu, Y. Yang, et al: Real-time Hand Gesture Recognition From Depth Images Using Convex Shape Decomposition Method J]. Journal of Signal Processing Systems, Vol. 74 (2014) No.1, p.47-58.

[11] C Y. Kao, C S. Fahn: A Human-machine Interaction Technique: Hand Gesture Recognition Based on Hidden Markov Models with Trajectory of Hand Motion [J]. Procedia Engineering, Vol. 15 (2011) No.1, p.3739-3743.

[12] H. Liu, W. Liu, L J. Latecki: Convex shape decomposition[C]// Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on. IEEE (San Francisco, CA, USA, June 13-18, 2010), p.97-104.

[13] N. Luo, F. Qian: Multi-objective Evolutionary of Distribution Algorithm Using Kernel Density Estimation Model[C]// Intelligent Control and Automation (WCICA), 2010 8th World Congress on. IEEE (July 7-9, 2010), p.2843-2848.

[14] X F. Yin, Z F. Hao: Fast Kernel Distribution Function Estimation and fast kernel density estimation based on sparse Bayesian learning and regularization[C]// 2008 International Conference on Machine Learning and Cybernetics. IEEE (July 12-15, 2008), p.1756-1761.

[15] J. Guan, P. Van Hese, J O. Nino-Castaneda, et al: Phd forum: Correlation Coefficient Based Template Matching for Indoor People Tracking[C]// Distributed Smart Cameras (ICDSC), 2012 Sixth International Conference on. IEEE (October 30-November 2, 2012), p.1-2.